2

WRDC-TR-90-3037
VOLUME II

WRIGHT RESEARCH & DEVELOPMENT CENTER

Multivariable Methods for the Design, Identification, and Control of Large Space Structures

Vol. II. Optimal and Sub-optimal Estimation Applied to Large Flexible Structures

S. Dumbacher

University of Cincinnati
Dept of Aerospace Engineering & Engineering Mechanics
Cincinnati, Ohio 45221-0070

July 1989

Final Report for the Period July 1987 to February 1989

Approved for public release; distribution unlimited.

FLIGHT DYNAMICS LABORATORY
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise as in any manner, as licensing the holder or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Victoria A. Tischler

VICTORIA A. TISCHLER
Project Engineer
Design & Analysis Methods Group

Nelson D. Wolf

NELSON D. WOLF, Technical Manager
Design & Analysis Methods Group
Analysis & Optimization Branch

FOR THE COMMANDER

Mark S. Ewing

MARK S. EWING, Maj, USAF
Chief, Analysis & Optimization Branch
Structures Division

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/FIBRA, Wright-Patterson AFB OH 45433-6553 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR-90-3037, Volume II
6a. NAME OF PERFORMING ORGANIZATION University of Cincinnati 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Flight Dynamics Laboratory (WRDC/FIBRA) Wright Research and Development Center
6c. ADDRESS (City, State, and ZIP Code) Cincinnati, Ohio 45221-0070	7b. ADDRESS (City, State, and ZIP Code) Wright-Patterson Air Force Base, Ohio 45433-6553
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F33615-86-C-3216
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO	WORK UNIT ACCESSION NO.
61102F	2302	N5	04

11. TITLE (Include Security Classification) Multivariable Methods for the Design, Identification, and Control of Large Space Structures - Vol II. Optimal and Sub-optimal Estimation Applied to Large Flexible Structures

12. PERSONAL AUTHOR(S)
S. Dumbacher

13a. TYPE OF REPORT Final Report	13b. TIME COVERED FROM 87 July TO 89 Feb	14. DATE OF REPORT (Year, Month, Day) 1989, July	15. PAGE COUNT 114

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Large Space Structures; Multi-Variable Control;
Control Structure Interaction ,

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report investigates the integrated control/structure design optimization problem for large flexible structures subject to stochastic disturbance forces. The full state feedback control of truss type structures is considered. The problem is formulated as a mathematical programming problem, and solved using a sequential approximations solution technique along with a scaling procedure. Due to the special nature of full state feedback control, it is shown that the controller can be assumed to be an LQR controller, where the control and state weightings are Lagrange multipliers from the mathematical programming optimality conditions. The design space is therefore effectively reduced to the truss element cross-sectional areas and these Lagrange multipliers, significantly reducing the computational effort required for solution. In this report, the solution algorithm is applied to the DRAPER I tetrahedral truss structure.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Victoria A. Tischler	22b. TELEPHONE (Include Area Code) 513-255-6992 22c. OFFICE SYMBOL WRDC/FIBRA

Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

FOREWORD

This final report was prepared by the University of Cincinnati, Cincinnati, Ohio for the Analysis and Optimization Branch (FIBR) of the Wright Research and Development Center. The work was performed under Contract F33615-86-C-3216 which was initiated under Project No. 2032, "Structures", Task No. N5, "Structural Dynamics and Controls". The objective of this contract was to develop mathematical algorithms for the integration of structural dynamics and controls with particular reference to the design of large space structures. This report, which is Vol I of a three volume final report, is entitled "Estimator Eigenvalue Placement in Positive Real Design". The program manager was Dr G. L. Slater of the Aerospace Engineering Department. He was supported by Mr M. D. McLaren a graduate student in the same department. In FIBR V. A. Tischler was the project monitor while Dr V. B. Venkayya initiated the program and provided overall program direction.

TABLE OF CONTENTS

LIST OF FIGURES

List Of Tables

Chapter 1

Introduction

1.1 Introduction

The structures of today are rapidly increasing in size and complexity compared to those of even a few years ago. Additionally, with monetary constraints equating to weight constraints, the structures of today are also becoming lighter than ever before. While "large flexible structures" is typically a term reserved for space structures, it can also include modern airframes and even buildings. The common denominator for large flexible structures is their inherently low damping and stiffness characteristics, which necessitate the use of active control systems for vibration and shape control.

In the past, the structure and its control system have been designed independently. Structural design and optimization, and control system design and optimization, have both been areas of separate research, each progressing vigorously along its own path. However, spurred on by recent proposals of new, large, highly constrained space structures, the question has arisen as to whether an integrated structural/control design procedure might not be more appropriate.

The main aim of this work is to investigate this problem, setting forth and demonstrating a design methodology with which practical integrated design problems can be solved. The remainder of this chapter is devoted to a historical overview of the control/structure interaction (CSI) problem, with respect to the problem formulation and the numerical solution techniques employed. Chapter 2 presents the general mathematical programming formulation of the design problem, then considers how the particular controller methodology under consideration modifies and adds to this probem formulation.

In Chapter 3, the specific numerical solution technique used in this work is presented. An algorithm is developed to solve the mathematical programming problem as set up in Chapter 2, and a numerical technique for the special case when full state feedback control is used is considered. In Chapter 4, the solution algorithm is illustrated on a representative structure — the DRAPER I Tetrahedral Truss structure. Finally, from these results some general conclusions about this problem are drawn, and recommendations for future topics of research given, in Chapter 5.

1.2 Historical background

The first papers actively investigating the question of simultaneous structure and control design began appearing in the literature around 1983 [Han, Kom, Hal-1]. Since then, there has been a growing interest in this subject from other authors, although the field itself is still in relative infancy. This section will attempt to review the current state of the art in the simultaneous structure/control optimization of large flexible structures. The section is broken into two main parts dealing with the general problem formulations presented and the numerical solution techniques employed. Although these are somewhat related, they are largely separate since several numerical methods could be used to solve each formulation, and several formulations could be solved using the same numerical procedure.

1.2.1 Problem Formulation

Most authors begin by assuming, if only implicitly, that the simultaneous design of the structure and the control system will definitely reap benefits that outweigh the increased computational expense that will inevitably follow. On the surface, this would seem a reasonable assumption, since a traditional sequential design process may involve several iterations by both the structural design group and the control synthesis group. However, Haftka et al. [Haf-1] found that, specifically in his case for the integrated thermal/structural design problem, integrating normally independent design methodologies does not always justify the effort and expense involved.

Therefore, to investigate the possible benefits of an integrated structure/control design approach, Haftka et al. [Haf-3] consider small changes in the structural parameters. They find that significant enhancements in the control can be obtained by these small changes. That is, small structural modifications can result in large control effort reductions to meet the same vibration control requirements. This result, and other recent studies also demonstrating synergistic effects of an integrated design (for example, [Kho-1,Sal,Mes]), are seen as a justification for further investigation into the problem.

Using a conventional design approach for a controlled structure, one would first optimize the structure alone, then design a control system for this baseline structure. This process may then be iterated until both the structure and control system meet necessary constraints and objectives. The structure may have constraints such as a maximum total mass and minimum open-loop frequencies and spacings, stress and/or displacement constraints under various loading conditions, and perhaps even specified open-loop mode shapes. The control system constraints may include closed-loop damping and frequency, frequency spacing, total available control energy, and maximum output response constraints. In this conventional approach then, the structure would be designed to minimize the total mass subject to open-loop structural constraints, and the controller would then be designed in some optimal fashion, perhaps as an linear quadratic gaussian (LQG) regulator, subject to the closed-loop constraints. The objective to be minimized in the control design would be a quadratic form of the type

$$\int_0^\infty \left\{ \boldsymbol{x}^T A \boldsymbol{x} + \boldsymbol{u}^T B \boldsymbol{u} \right\} dt \quad \text{or} \quad E\left[\boldsymbol{x}^T A \boldsymbol{x} + \boldsymbol{u}^T B \boldsymbol{u} \right] \tag{1.1}$$

Therefore, a "classical" approach to the simultaneous structure/control optimization would be to simultaneously minimize the weighted sum of the total mass and the above quadratic form, subject to all of the structural and control constraints. That is, define the objective as

$$J(\boldsymbol{p}) = a_1 W(\boldsymbol{p}) + a_2 \int_0^\infty \left\{ \boldsymbol{x}^T A(\boldsymbol{p}) \boldsymbol{x} + \boldsymbol{u}^T B(\boldsymbol{p}) \boldsymbol{u} \right\} dt \tag{1.2}$$

There are a number of authors who formulate the integrated control/structure design optimization using this "classical" approach. Salama, Hamidi and Demsetz [Sal], Messac, Turner and Soosaar [Mes], Miller and Shim [Mil-1] and Onoda and Haftka [Ono] are members of this group. In addition, these authors simplify the problem by considering that, for any fixed parameter vector $\boldsymbol{p}$, the optimal constant gain control is given by the well known Linear Quadratic Regulator (LQR) solution. Then, equation (1.2) reduces to

$$J(\boldsymbol{p}) = a_1 W(\boldsymbol{p}) + a_2 \boldsymbol{x}^T(\boldsymbol{p}) P_{ss}(\boldsymbol{p}) \boldsymbol{x}(\boldsymbol{p}) \tag{1.3}$$

where P_{ss} is the steady-state solution to the appropriate Riccati matrix equation. This significantly reduces the dimension of $\boldsymbol{p}$ since the control gains are no longer included in the optimization, and can result in a significant computational savings.

Hale, Lisowski and Dahl [Hal-2] also define an objective function of the form of (1.3), and augment it with functions of auxiliary coordinates, Lagrange multipliers and adjoint displacements. Necessary conditions for an optimal control at fixed $\boldsymbol{p}$, along with necessary conditions for an optimal $\boldsymbol{p}$, can then be derived by the calculus of variations. The resulting coupled nonlinear equations must be solved numerically due to their general complexity.

As noted by Hale and Lisowski [Hal-3], one of the major problems in assuming a quadratic controller penalty function ot the form of (1.1) is the difficulty in assigning a meaningful relative penalty on the deflection vs. the penalty on the control energy. This same problem exists in the separate optimal control problem, but is exacerbated in the integrated design problem since the structural parameters are varied. Additionally, we must also assign a relative weight to the structural mass part of the combined objective (1.2) vs. the control part of (1.2), i.e. we must assign values to a_1 and a_2 in equation (1.2). The optimal design so calculated will be very sensitive to the values of these weighting parameters, but present methods rely more on artwork to choose these weights than on any formalized algorithm. One way to avoid choosing these weights would be to extremize only one part of the joint objective function (1.2), and include the effects of the other part explicitly into the constraints.

In [Hal-3], the authors use the total maneuver control cost as the objective to be minimized, subject to initial and final displacement and velocity constraints (maneuver constraints). Khot, Eastep and Venkayya [Kho-1], Khot [Kho-2], Navid [Nav] and Manning and Schmit [Man] all consider the problem of integrated control/structure design as a mass minimization subject to constraints on the closed-loop system response (displacement and/or eigenvalue constraints). While the control cost can easily be a function of structural parameters through the system matrices, it is usually assumed that the total mass is not a function of controller parameters (i.e. neglecting the mass of the actuators and their energy source). Therefore, when considering a minimum mass design, it is the interaction with the constraints on the closed-loop system that provides the simultaneous design.

Bodden and Junkins [Bod] seek to minimize the norm of a vector of output feedback gain matrix elements to "herd" the closed-loop eigenvalues into an acceptable region of the complex plane. In this sense, the norm of this vector was taken as a measure of the control effort. In [Jun], Junkins and Rew extend the ideas presented in [Bod] to cases where the controller is designed as a classical LQG regulator. Here, the elements of the weight matrices in the quadratic performance index are included in the parameter vector p, and the authors show that not all weight matrices are created equal.

Haftka, et al. [Haf-3,Haf-2] choose a similar objective function, in this case the sum of the gains of the actuators. Since in their case the actuators act as viscous dashpots attached to a flexible bar, the sum of these gains gives a measure of the total amount of damping supplied by the actuators and colocated velocity sensors. In this sense, the sum of the gains is also a measure of control effort.

Rather than minimize the sum of the structural mass and a control performance term, Lim and Junkins [Lim-1] instead review three measures of closed-loop stability robustness which could be extremized. This paper is essentially a condensed version of the Ph.D. dissertation of Lim [Lim-2]. Large flexible structures have many closely spaced, lowly damped, low frequency modes, and since present finite element techniques will only accurately calculate the first few modes, there may be considerable uncertainty in modal frequencies and mode shapes within the bandwidth of the controller. Therefore, the closed-loop system should be as robust as possible to variations in these parameters.

The first robustness measure is an eigenvalue sensitivity norm, which is the maximum singular value (L_2 norm) of a matrix product representing an upper bound on the square of a weighted eigenvalue error norm for normalized perturbations p. This measure would be minimized to give the structure stability robustness. Another measure is the well- known condition number, which can be related to the radius of uncertainty within which all eigenvalues are perturbed due to an error in the system matrix. At the very least, we would like the closed-loop system to remain asymptotically stable in the presence of parameter uncertainties, and this corresponds to maximizing the stability robustness. It is shown that this is equivalent to minimizing the condition number of the closed-loop system matrix.

The third robustness measure is due to Patel and Toda [Pat], and is related to the spectral radius of the solution of an associated Lyapunov equation. This measure should be maximized. The major problem with measures of robustness is their conservatism. However, the authors in [Lim-1] conclude from their numerical studies that a significant improvement in the robustness measure does indeed correspond to a significant improvement in the actual stability robustness of the closed-loop system.

Adamian [Ada] chooses another measure of robustness, and formulates the problem as a minimization of a composite objective function which is a linear combination of the structural weight and the robustness measure. In his work, the sensitivity of the closed-loop eigenvalues with respect to plant uncertainties is taken as the measure of robustness. Closed-loop eigenvalue constraints are also included.

Gustafson, et al. [Gus] set the problem up in a unique fashion. They have developed a method to obtain equivalent continuum models for truss type structures with a pattern of repeating elements. From this equivalent continuum model (discrete modelling is used for rigid components of the structure), PDE's are obtained and solved in the frequency domain so that explicit dependence on structural parameters is achieved. The solution is in distributed form, which thus retains all modal information available in the PDE. The sets of structural and control parameters can be combined into a single optimization problem - the simultaneous control/structure design problem.

Hale [Hal-4], proposes an approach to the simultaneous synthesis for vibration suppression using the set theoretic analysis method for linear dynamic systems. First, external disturbances are assumed to be unknown but bounded time-varying processes. Second, inequality constraints on outputs (measurements) and inputs (controls) are included explicitly. The purpose of the integrated synthesis is to maximise the amplitude bound on the unknown disturbances while explicitly satisfying input and output constraints.

A similar approach is taken by Slater [Sla], in that no explicit relative weight between the structural parameters and the control terms is required. Rather, the control is viewed as complementing the effect of the structure in reducing deformations due to an applied stochastic disturbance model. The formulations by Manning and Schmit [Man] and Lust [Lust] are similar except that the excitation forces are assumed deterministic and known, and are explicitly harmonic in the case of Lust. Thus, in all members of this last group, real physical constraints drive the solution rather than the vague quadratic penalties in a cost functional.

One thing to keep in mind concerning the preceeding discussion is that the choice of objective to be extremized will usually be highly mission dependent. The problem formulation will always involve some element of engineering judgement.

1.2.2 Problem Solution

Once the problem has been formulated as a nonlinear programming problem, their solution presents a challenge. There is a critical need for a dependable optimization tool that efficiently handles a large number of nonlinear inequality constraints and a very high dimensional design space. In the literature, many numerical solution approaches are presented, almost all of which are gradient based. All solution techniques are iterative due to the nonlinearity of the problems to be solved, and may only converge to local minima due to the fact that the objective function will usually not be globally convex. In fact, this problem of convergence to local minima is inherent to gradient based approaches, and generally cannot be avoided.

The gradients of the objective and constraint functions with respect to the design parameters will be required. For highly nonlinear and large dimensionality problems, numerically evaluating these gradients using finite differencing can be computationally expensive, and the gradients may include a significant error over their analytically evaluated counterparts. Such errors may halt the solution algorithm prematurely. Therefore, if at all possible, efficient analytical expressions for the gradients should be derived. Even so, convergence rates may be slow, hence numerical solutions for even relatively simple structures may still be computationally expensive.

For algebraic optimization, where there are no differential type constraints, there are many commercially available nonlinear programming codes that can be used to solve general nonlinear constrained optimization problems. Haftka, et al. [Haf-3, Haf-2] used NEWSUMT [Miu], which employs an extended interior penalty function formulation using Newton's method with approximate second derivatives for each constrained minimization. NEWSUMT-A [Gra-1] was developed from NEWSUMT by adding constraint approximations and a move-limit strategy, and is used by Onoda and Haftka [Ono].

The VMCON optimization subroutine [Cra], which is based on Powell's algorithm for nonlinear constraints that uses Lagrangian functions, was used by Khot, Eastep and Venkayya [Kho-1] and Khot [Kho-2]. Gustafson, et al. [Gus] used the IMSL subroutine ZXSSQ [IMSL]. This subroutine finds the least-squares solution based on a modification of the Levenberg- Marquardt algorithm which eliminates the need for explicit derivatives. This is a special case because the example considered in [Gus] had no constraints. ZXSSQ can not be used for the solution of constrained optimization problems.

Manning and Schmit [Man] used the subroutine CONMIN [Van-1] to solve an approximate problem that is derived from the nonlinear optimization problem. CONMIN was also used by Lust [Lust] and Navid [Nav] to solve their sequences of approximate problems. A mathematical programming program that includes CONMIN, as well as a variety of other optimizing algorithm options, is ADS [Van-3]. This was used by Adamian [Ada] to directly solve his optimization problem. Both CONMIN and ADS are public domain codes, widely used because of their generality and ease of application.

Salama, Hamidi and Demsetz [Sal] propose an iterative solution, based on a modified Newton-Raphson scheme, of the appropriate Kuhn-Tucker optimality conditions. However, this solution requires the calculation of the second derivatives of the objective function with respect to the design variables. For high order systems especially, evaluating these second derivatives becomes very costly. In order to avoid this expense, Miller and Shim [Mil-1] explore the use of the method of steepest descent to minimize the objective. If no constraints are active, this method is the same as the gradient projection method mentioned above. When constraints are active, the objective is augmented by these active constraints using Lagrangian multipliers, and this augmented objective is then minimized by steepest descent as before.

Rather than solving the full nonlinear programming problem, some authors prefer to work with linearized versions of the full nonlinear equations. In this sort of iterative algorithm, move limits must be placed on the design variables in order that the local linearity be satisfied. Bodden and Junkins [Bod] and Junkins and Rew [Jun] linearize the objective and constraint functions about the current value of the design variables. The initial design point is taken as the output of a typical sequential design process. The authors adopt a "minimum modification" strategy, whereby from the infinity of possible solutions to a small number of equations (linearized objective and constraints) in a large number of unknowns (elements of the design variable vector), the solution is chosen that minimizes the changes in the design variables while achieving the desired changes in the objective and constraints.

The numerical proceedure used here is equivalent to a gradient projection constrained optimization algorithm. Convergence of the iterative algorithm can be enhanced by employing a continuation proceedure, whereby a new parameter is introduced to allow the creation of "stepping stone" problems that are arbitrarily close to the converged neighbouring solutions [Bod,Hor]. The continuation approach allows one to almost arbitrarily satisfy the local linearity requirements by choosing a small enough step in the continuation parameter. It also allows, for example, restrictive constraints to be applied gradually from less restrictive ones.

Lim and Junkins [Lim-1] propose an approach to the solution of nonlinear programming problems involving sequential linear programming. Here, the nonlinear problem is transformed into the standard linear programming problem by linearization, a translational transformation of the design parameters, and by adding slack and/or surplus variables. Once in this standard linear format, the problem formulation is straightforward, and very efficient and reliable linear programming software is available for its solution.

Hale [Hal-4] introduces the use of set-theoretic methods for the solution of integrated structure/control synthesis problems. Due to the author's unique problem formulation mentioned earlier, the constraints on the output and control terms are box-type constraints, which can be approximated (conservatively) as ellipsoidal sets. Set-theoretic methods can then be applied to maximize the magnitude of the disturbances such that the ellipsoidal set constraints are not violated. However, the author acknowledges that the computational burden of a numerical solution using this method is very high. This would prohibit solving problems of actual structures, with a large number of states and/or design parameters, using this method.

For problems with differential or integral constraints, a calculus of variations approach to the problem solution yields, through the Fundamental Theorem, the necessary conditions to be satisfied by an extremal [Kir-1]. These necessary conditions are the Euler equations, and in general are nonlinear two- point boundary value differential equations. Generally, these problems will be much more difficult to solve than a corresponding algebraic optimization problem. In fact, solving these equations in an iterative framework can become prohibitively expensive [Hal-2].

One method of solving the two-point boundary value problems, that uses a projected gradient iterative method, is given in [Hal-2]. Similar techniques are also used by Messac, Turner and Soosaar [Mes] and by Hale and Lisowski [Hal-3] for solution of their integrated design problem formulation. Basically, the iterations proceed in the negative direction of the gradient of the objective (direction of greatest decrease of objective) until a constraint is met. At this point, the gradient vector is projected onto the constraint, and this becomes the direction of motion for that iteration. The comment is made however, that these algorithms are not yet efficient enough to tackle large problems [Hal-3].

1.3 Scope of the Present Work

The work contained in this report is intended to extend the work of Slater [Sla], in which a preliminary study of control/structure optimization was performed. Here, the optimization will be based on the dynamic response of a structure to an external disturbance environment, and as such will differ from most previous work. Such a response to excitation approach is common to both the structural and control design phases, and hence represents a more natural control/structure optimization strategy than relying on the artificial and vague control penalties used by other authors. The disturbances are to be considered unknown and stochastic, and can therefore model a wide variety of actual disturbance states. The design objective is to find the structure and controller of minimum mass such that all the prescribed constraints are satisfied. The controller interaction will be inserted by the imposition of appropriate closed-loop constraints, such as closed-loop output response and control effort constraints.

Chapter 2

The Integrated Control/Structure Optimization Problem

2.1 Introduction

In this chapter, the integrated control/structure design optimization problem is formulated as a general nonlinear constrained mathematical programming problem. This problem formulation is quite general with respect to thc constraints that can be applied and the controller methodology that can be employed. In this work, only the special case of full state feedback control will be considered. This enables the problem to be partially solved analytically, and substantially reduces the dimensionality of the numerical problem to be solved. In future work, more realistic and/or sophisticated controller methodologies will be considered.

2.2 General Problem Formulation

The integrated control/structure design optimization problem can be stated as follows: find the vector of structural and controller parameters that minimizes the mass of the structure subject to limitations on the available control energy, and a set of allowable output responses to a set of prescribed stochastic disturbances. Mathematically, this can be written in the form of a nonlinear mathematical programming problem as

Minimize, with respect to $\boldsymbol{p}$, the weight

$$J(\boldsymbol{p})$$

subject to:

$$g_j(\boldsymbol{p}) \leq 0 \qquad \text{for } j = 1, \ldots, m$$

$$\dot{\boldsymbol{x}} = F\boldsymbol{x} + G\boldsymbol{u} + G_w\boldsymbol{w}$$

$$g_{cc_i} = \frac{\mathrm{E}[\boldsymbol{u}_i^T R_i \boldsymbol{u}_i]}{\beta_i^2} - 1 \leq 0 \qquad \text{for } i = 1, \ldots, n_\beta \tag{2.1}$$

$$g_{oc_i} = \frac{\mathrm{E}[\boldsymbol{y}_{d_i}^T W_i \boldsymbol{y}_{d_i}]}{\alpha_i^2} - 1 \leq 0 \qquad \text{for } i = 1, \ldots, n_\alpha$$

$$\boldsymbol{y}_{d_i} = H_{d_i}\boldsymbol{x} \qquad \text{for } i = 1, \ldots, n_\alpha$$

$$\boldsymbol{p}_\ell \leq \boldsymbol{p} \leq \boldsymbol{p}_u$$

where

$\boldsymbol{p}$ is an N-vector of design variables,

g_j is the j^{th} element of $\boldsymbol{g}$, an m-vector of structural constraints,

$\boldsymbol{x}$ is an n-vector of state variables,

$\boldsymbol{u}$ is an n_u-vector of control forces,

$\boldsymbol{w}$ is an n_w-vector of stochastic disturbances,

F is the $(n \times n)$ matrix containing the system dynamics,

G is the $(n \times n_u)$ matrix containing information on the locations and orientations of the actuators,

G_w is the $(n \times n_w)$ matrix containing information on the points of application and orientation of the disturbances,

g_{cc_i} is the i^{th} control effort constraint cost function,

$\boldsymbol{u}_i$ is that n_{u_i}-order partition of $\boldsymbol{u}$ representing the control forces involved in g_{cc_i},

β_i is the maximum allowable value of the i^{th} expected control effort function $\mathrm{E}[\boldsymbol{u}_i^T R_i \boldsymbol{u}_i]$,

R_i is an $(n_{u_i} \times n_{u_i})$ control force weighting matrix,

g_{oc_i} is the i^{th} output response constraint cost function,

α_i is the maximum allowable value of the i^{th} expected output response function $\mathrm{E}[\boldsymbol{y}_{d_i}^T W_i \boldsymbol{y}_{d_i}]$,

W_i is an $(n_{d_i} \times n_{d_i})$ output response weighting matrix,

$\boldsymbol{y}_{d_i}$ is the i^{th} design output n_{d_i}-vector,

H_{d_i} is an $(n_{d_i} \times n)$ matrix giving the relationship between the state variables and $\boldsymbol{y}_{d_i}$,

$\boldsymbol{p}_\ell$ is an N-vector of minimum design variable values, and

$\boldsymbol{p}_u$ is an N-vector of maximum design variable values.

The side constraints are the strict bounds $\boldsymbol{p}_\ell$ and $\boldsymbol{p}_u$ on the design variables, and are vector inequalities that are imposed element by element. These design variable bounds are not included *explicitly* as *constraints* in the problem formulation. Note that the structural weight and the structural constraints $\boldsymbol{g}$ will in general not be functions of the controller design variables unless the controller mass is included in the design. Note also that g_{cc_i} is a weighted mean square control effort, and g_{oc_i} is a weighted mean square output response. Multiple output response constraints are allowed, although only one of these will in general be active at the optimum design. However, all of the control effort constraints will generally be active at the optimum design.

The disturbances acting on the structure, represented by $\boldsymbol{w}$, can conceivably be in any form one desires. However, it is assumed in this work that $\boldsymbol{w}$ is a stochastic disturbance, specifically zero mean Gaussian white noise with covariance X_w. The structure will respond to this disturbance with some transient behaviour, in addition to a steady-state response. It seems reasonable to optimize the structure for the steady-state response to the disturbance rather than the transient response because the transient behaviour will normally be of secondary importance to the response objectives (such as long term pointing accuracy). Also, for steady state optimization, the differential equation constraint (state equation) can be replaced with a steady state covariance equation, and the mean square control effort and output response constraints may be recast in terms of this covariance. Therefore the two-point boundary value problem is eliminated and the numerical solution of the problem is significantly simplified.

The particular form of this covariance equation will depend on the form of the controller used, and will be derived for each controller methodology as they are presented. Each covariance equation will be of the general form

$$F_{cl}X + XF_{cl}^T + Q = 0 \tag{2.2}$$

where F_{cl} is the closed-loop dynamical system matrix, X is the symmetric positive definite covariance matrix, and Q is a symmetric positive definite or semi-definite matrix.

To obtain the first-order necessary conditions for optimality, first form the augmented objective function J_a as

$$\begin{aligned} J_a &= J(\boldsymbol{p}) + \mathrm{tr}\left[\Lambda_x(F_{cl}X + XF_{cl}^T + Q)\right] + \sum_{j=1}^{n_\beta} \lambda_{u_j} g_{cc_j}(\boldsymbol{p}) + \\ &\qquad + \sum_{j=1}^{n_\alpha} \lambda_{y_j} g_{oc_j}(\boldsymbol{p}) + \sum_{j=1}^{m} \mu_j g_j(\boldsymbol{p}) \end{aligned} \tag{2.3}$$

Then the Kuhn-Tucker necessary conditions for a constrained minimum state that, at the optimum design, the following conditions must be satisfied [Van-2]:

$$\frac{\partial J_a(\boldsymbol{p})}{\partial p_i} \begin{cases} > 0 & \text{if } p_i = p_{\ell_i} \\ = 0 & \text{if } p_{\ell_i} < p_i < p_{u_i} \\ < 0 & \text{if } p_i = p_{u_i} \end{cases} \qquad \text{for } i = 1, \ldots, N \tag{2.4}$$

$$\begin{aligned} \lambda_{u_i} &\geq 0 \quad \text{for } i = 1, \ldots, n_\beta \\ \lambda_{y_i} &\geq 0 \quad \text{for } i = 1, \ldots, n_\alpha \\ \mu_j &\geq 0 \quad \text{for } j = 1, \ldots, m \end{aligned} \tag{2.5}$$

$$\begin{aligned} \lambda_{u_j} g_{cc_j}(\boldsymbol{p}) &= 0 \quad \text{for } j = 1, \ldots, n_\beta \\ \lambda_{y_j} g_{oc_j}(\boldsymbol{p}) &= 0 \quad \text{for } j = 1, \ldots, n_\alpha \\ \mu_j g_j(\boldsymbol{p}) &= 0 \quad \text{for } j = 1, \ldots, m \end{aligned} \tag{2.6}$$

$$\begin{gathered} g_j(\boldsymbol{p}) \leq 0 \quad \text{for } j = 1, \ldots, m \\ g_{oc_j} \leq 0 \quad \text{for } j = 1, \ldots, n_\alpha \\ g_{cc_j} \leq 0 \quad \text{for } j = 1, \ldots, n_\beta \\ F_{cl}X + XF_{cl}^T + Q = 0 \\ \boldsymbol{p}_\ell \leq \boldsymbol{p} \leq \boldsymbol{p}_u \end{gathered} \tag{2.7}$$

In equations (2.4) - (2.8) above, Λ_x is the matrix of Lagrange multipliers corresponding to the covariance equality constraint equation (2.2), and $\boldsymbol{\lambda}_u$, $\boldsymbol{\lambda}_y$ and $\boldsymbol{\mu}$ are the vectors of the Lagrange multipliers corresponding to the inequality constraints of equations (2.2). Since the covariance Lyapunov equation (2.2) is symmetric, we can assume that Λ_x is symmetric also.

Equations (2.4) indicate that, at the optimum design, the gradient of the objective function augmented with the constraints via Lagrange multipliers must vanish for those variables not at their side constraints. This derivative can be written as

$$\begin{aligned} \frac{\partial J_a(\boldsymbol{p})}{\partial p_i} &= \frac{\partial J(\boldsymbol{p})}{\partial p_i} + \frac{\partial}{\partial p_i}\text{tr}[\Lambda_x(F_{cl}X + XF_{cl}^T + Q)] + \sum_{j=1}^{n_\beta} \lambda_{u_j} \frac{\partial g_{cc_j}(\boldsymbol{p})}{\partial p_i} + \\ &+ \sum_{j=1}^{n_\alpha} \lambda_{y_j} \frac{\partial g_{oc_j}(\boldsymbol{p})}{\partial p_i} + \sum_{j=1}^{m} \mu_j \frac{\partial g_j(\boldsymbol{p})}{\partial p_i} \quad \text{for } i = 1, \ldots, N \end{aligned} \tag{2.8}$$

These gradients are taken with respect tc all structural and controller design parameters. Equations (2.6) indicate that the Lagrange multipliers must be nonnegative at the optimum design. For those inequality constraints not active at the optimum, equations (2.7) indicate that the corresponding Lagrange multipliers must be zero. This is because non-active inequality constraints should take no part in the final design. Finally, and obviously, the optimum design should satisfy all constraints, as indicated by equations (2.8).

Now, consider how the general formulation as presented is affected by the particular choice of controller methodology. The controller constraints take on particular functional forms, as does the covariance equation.

2.3 Full State Feedback Control

The simplest form of feedback control is to feedback the entire state vector, as in

$$\boldsymbol{u} = -K\boldsymbol{x} \tag{2.9}$$

where K is the $(n_u \times n)$ state feedback gain matrix. The controller design variables for this case will be the $n_u n$ elements of K. A block diagram giving the structure of this controller is given in Figure 2.1. Substituting this control into the state equation, and assuming that the disturbance $\boldsymbol{w}$ is zero mean Gaussian white noise, the state covariance matrix X for this case can be found from the Lyapunov equation

$$F_{cl}X + XF_{cl}^T + G_w X_w G_w^T = 0 \tag{2.10}$$

where $F_{cl} = (F - GK)$ is the stable closed-loop dynamical matrix for the full state feedback case, $X = \mathrm{E}[\boldsymbol{x}\boldsymbol{x}^T]$ is the $(n \times n)$ symmetric state covariance matrix, and $X_w = \mathrm{E}[\boldsymbol{w}\boldsymbol{w}^T]$ is the $(n_w \times n_w)$ symmetric covariance matrix for the stochastic disturbances.

Expressions for the controller constraints in terms of this covariance matrix can then be obtained as

$$g_{cc_i} = \frac{\text{tr}[K_i^T R_i K_i X]}{\beta_i^2} - 1 \qquad \text{for } i = 1, \ldots, n_\beta \tag{2.11}$$

$$g_{oc_i} = \frac{\text{tr}[H_{d_i}^T W_i H_{d_i} X]}{\alpha_i^2} - 1 \qquad \text{for } i = 1, \ldots, n_\alpha \tag{2.12}$$

where K_i is the $(n_{u_i} \times n)$ partition of K corresponding to $\boldsymbol{u}_i$. It is assumed that the $\boldsymbol{u}_i$ are independent, and that $\boldsymbol{u}$ is ordered as

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_{n_\beta} \end{bmatrix} \tag{2.13}$$

Note that $\sum_{i=1}^{n_\beta} n_{u_i} = n_u$, and that the columns of G can be interchanged to force condition (2.13) to be satisfied. The elements of the feedback gain matrix K, along with the elements of the covariance matrix X, are the independent controller design variables in the augmented Lagrangian approach. The introduction of the Lagrange multipliers means that all variables are to be considered independent, even though X appears dependent on a particular choice of K through equation (2.10).

Consider equations (2.8) when the gradients are taken with respect to the elements of the gain matrix K. To simplify notation, consider a matrix A with elements a_{ij}, and define for scalar s the matrix $\frac{\partial s}{\partial A}$ by

$$\left[\frac{\partial s}{\partial A}\right]_{ij} = \frac{\partial s}{\partial a_{ij}} \tag{2.14}$$

Using this notation, the Kuhn-Tucker equations for differentials with respect to K can be written

$$\frac{\partial}{\partial K}\left[\text{tr}\left(\Lambda_x[(F - GK)X + X(F - GK)^T]\right)\right] +$$

$$+ \frac{\partial}{\partial K}\left[\sum_{i=1}^{n_\beta} \lambda_{u_i}\left(\frac{\text{tr}[K_i^T R_i K_i X]}{\beta_i^2} - 1\right)\right] = 0 \tag{2.15}$$

Let R be an $(n_u \times n_u)$ matrix defined in a block diagonal sense as

$$R = \operatorname{diag}\left\{\left(\frac{\lambda_{u_i}}{\beta_i^2}\right) R_i\right\} \tag{2.16}$$

then equation (2.15) can be rewritten as (since $\boldsymbol{\lambda}_u$ is not a function of K)

$$\frac{\partial}{\partial K}\left[\operatorname{tr}\left(\Lambda_x[(F-GK)X + X(F-GK)^T]\right) + \operatorname{tr}[K^T R K X]\right] = 0 \tag{2.17}$$

Using the differentiation rules for the trace operator given in Appendix A, equation (2.17) becomes

$$-2G^T\Lambda_x X + 2RKX = 0 \tag{2.18}$$

which, since X is non-singular, and if $\lambda_{u_i} \neq 0$ for all i, reduces to

$$K = R^{-1}G^T\Lambda_x \tag{2.19}$$

When the gradients are taken with respect to the elements of X, equations (2.8) become

$$\frac{\partial}{\partial X}\left[\operatorname{tr}\left(\Lambda_x[(F-GK)X + X(F-GK)^T]\right) + \sum_{i=1}^{n_\beta}\lambda_{u_i}\left(\frac{\operatorname{tr}[K_i^T R_i K_i X]}{\beta_i^2} - 1\right)\right] + \\ + \frac{\partial}{\partial X}\left[\sum_{i=1}^{n_\alpha}\lambda_{y_i}\left(\frac{\operatorname{tr}[H_{d_i}^T W_i H_{d_i} X]}{\alpha_i^2} - 1\right)\right] = 0 \tag{2.20}$$

Equation (2.20), along with the definition of K in equation (2.19) and that of R in equation (2.16), gives

$$F^T\Lambda_x + \Lambda_x F - \Lambda_x G R^{-1} G^T \Lambda_x + W = 0, \tag{2.21}$$

where

$$W = \sum_{i=1}^{n_\alpha} \left(\frac{\lambda_{y_i}}{\alpha_i^2} \right) H_{d_i}^T W_i H_{d_i}, \tag{2.22}$$

which is an algebraic Riccati equation. Equations (2.19) and (2.21) define the solution to the optimal control problem

$$\min \; J_c = \int_0^\infty \left[\boldsymbol{x}^T W \boldsymbol{x} + \boldsymbol{u}^T R \boldsymbol{u} \right] dt \tag{2.23}$$

where K is the optimal steady-state gain matrix, and Λ_x is the steady-state solution to the associated Riccati equation. Sufficient conditions for the existance of a steady-state Riccati equation solution and stability of the control law given by equation (2.19), is that the system be completely controllable and detectable. Detectability indicates that all unstable modes are observable. If these mildly restrictive conditions are met, the closed-loop system F_{cl} is guaranteed to be asymptotically stable.

Therefore, at the optimum design, where the Kuhn-Tucker optimality necessary conditions for a minimum will be satisfied, the control design variables will be the solution of the linear quadratic regulator (LQR) problem equation (2.23). Although this LQR property only holds true at the optimum point, it will be assumed that at every point in the design cycle, the control design variables will be found as the solution to the optimal control problem (2.23). Therefore, the numerical optimization problem can be reduced to optimization over just the structural design variables, along with an optimal control problem solution which will be a function of the Lagrange multipliers $\boldsymbol{\lambda}_u$ and $\boldsymbol{\lambda}_y$. The immediate benefit of this is a reduced dimensionality nonlinear programming problem. In addition since under the conditions stated above, the regulator solutions always give a stable closed-loop system, no explicit check must be performed on the system stability during the solution procedure.

In order to considerably improve the conditioning of the optimization problem, normalize the objective function by its initial value — the initial weight of the structure W_0 [Van-2]. If the structural constraints as given in $\boldsymbol{g}$ are normalized, and since the controller constraint functions have been written in normalized form, the optimization objective and constraint functions will then all be of order one.

Finally then for this case, if $\boldsymbol{p}^s$ is the vector partition of $\boldsymbol{p}$ representing the *structural* design variables only, the integrated full state feedback control/structure design optimization problem can be stated as follows:

Minimize, with respect to $\boldsymbol{p}^s$, the normalized weight

$$\frac{1}{W_0}J(\boldsymbol{p}^s)$$

subject to:

$$g_j(\boldsymbol{p}^s) \leq 0 \qquad \text{for } j = 1, \ldots, m$$

$$g_{cc_i} = \frac{\text{tr}[K_i^T R_i K_i X]}{\beta_i^2} - 1 \leq 0 \qquad \text{for } i = 1, \ldots, n_\beta$$

$$g_{oc_i} = \frac{\text{tr}[H_{d_i}^T W_i H_{d_i} X]}{\alpha_i^2} - 1 \leq 0 \qquad \text{for } i = 1, \ldots, n_\alpha \tag{2.24}$$

$$F_{cl}X + XF_{cl}^T + G_w X_w G_w^T = 0$$

$$K = R^{-1}G^T\Lambda_x$$

$$F^T\Lambda_x + \Lambda_x F - \Lambda_x G R^{-1} G^T \Lambda_x + W = 0$$

$$\boldsymbol{p}_\ell \leq \boldsymbol{p} \leq \boldsymbol{p}_u$$

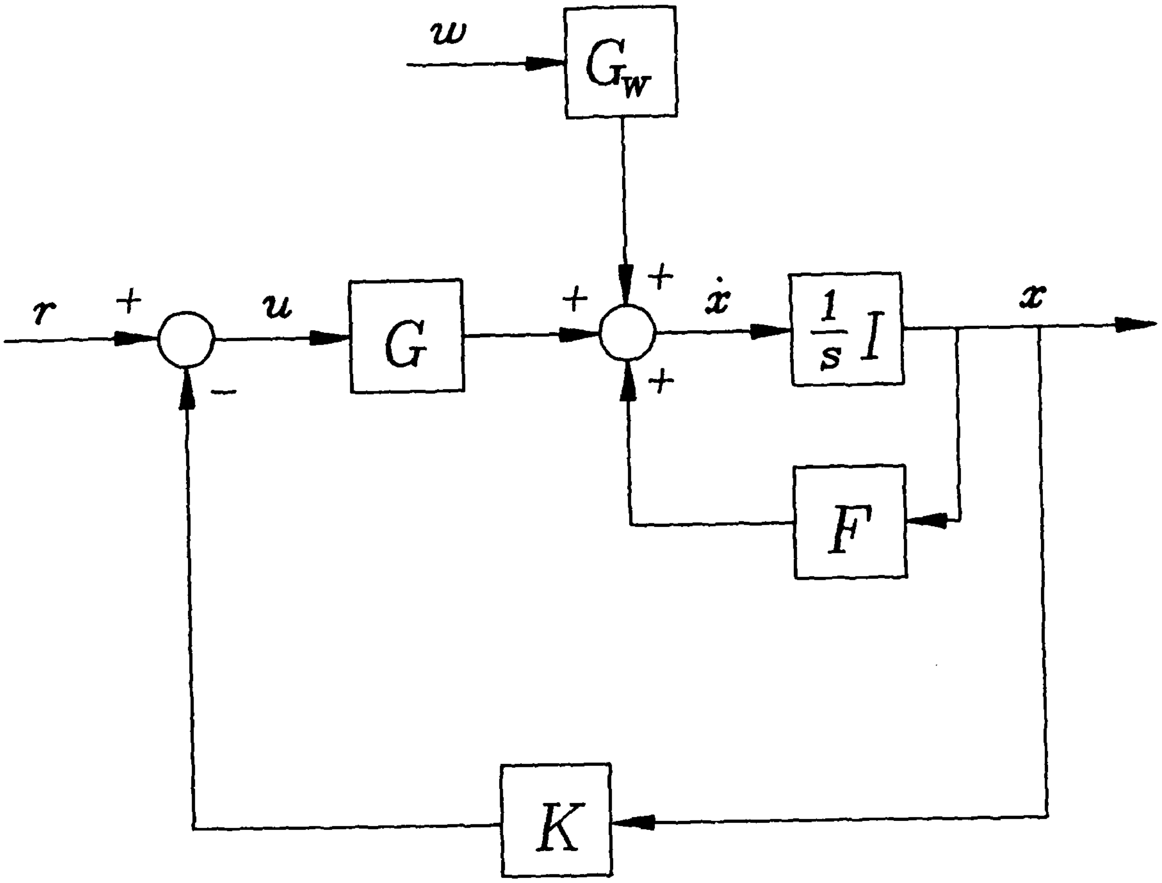

Figure 2.1: Full State Feedback Block Diagram

Chapter 3

Numerical Solution Technique

3.1 Introduction

In the general problem formulation presented in Chapter 2, the constraint functions are generally highly nonlinear implicit functions of the design variables. Solution of this problem could be attempted by the direct application of nonlinear programming techniques; that is, using the exact functional expressions for the constraints. However, this approach quickly becomes computationally very expensive as the dimensionality increases since the full objective and constraint functions must be evaluated at every step, and their respective gradients at most, if not all, steps thoughout the design procedure. Such evaluations tend to be computationally very expensive.

Approximation techniques, where the implicit nonlinear problem is replaced by a sequence of explicit approximate (although not necessarily linear) problems, have been shown to yield efficient and powerful algorithms for structural design optimization (see, for example, [Sch,Gra-2]). It has yet to be shown however, that the sequential approximations technique can be applied successfully to the integrated control/structure design optimization problem. Difficulties arise because in general, the control constraint functions tend to be much more complex, implicit nonlinear functions of the design variables than do the purely structural constraint functions. This means that to use approximation techniques in the integrated control/structure optimization, either much smaller steps in the design space will be required, or a more sophisticated approximation scheme must be implemented.

3.2 Solution Algorithm

In this method, the fully constrained nonlinear optimization problem is solved by the iterative construction and numerical solution of a sequence of explicit approximate problems. The approximate problems are basically first-order Taylor's series expansions (in some convenient intermediate variables), of the objective and constraint functions. The numerical solution is accomplished using the method of feasible directions as implemented in the mathematical programming code ADS [Kir-1]. We still solve the approximate problems using a nonlinear programming code because, depending on the intermediate variables chosen, the approximate functions may still be nonlinear functions of the design variables.

A schematic of the solution algorithm is shown in Figure 3.1. The solution process begins with some initial structure, which is analyzed using the finite element technique. At this point, the gradients of the active constraint set are evaluated, and the approximate problem is formed, with respect to the current design. Expressions for the gradients of all constraints considered can be evaluated analytically. The gradients for constraints common to all controller types, along with those introduced for specific controller methodologies, are derived in a later section.

The approximate problem is solved with ADS using an active constraint set strategy to reduce the dimensionality of the approximate problem by deleting the inactive constraints. Move-limits on the design variables are imposed during the solution to ensure that the design remains within the region for which the approximation functions are of acceptable quality. The choice of move-limits and how they change can have a significant effect on convergence, and will often be determined from numerical experience with the particular problem at hand.

After the solution of the approximate problem, the structure and its control system are deemed optimal if a convergence test on either the absolute or relative objective function change over a specified number of successive global iterations is satisfied. Otherwise, the objective and active constraint gradients are evaluated for the new design, a new approximate problem formed, and the process above is repeated in an iterative manner. The solution procedure ends when the design variables converge, or when the number of iterations exceeds some preset maximum.

Scaling the structure and controller to the closest constraint surface is possible in the case discussed in this work, because of the special assumed form of the controller. This scaling is performed on the initial system, and on the system immediately following the solution to the approximate problem. The scaling procedure is discussed in a later section. The solution procedure outlined above has been implemented in a research computer program CSOPT, written in FORTRAN, and run on a CRAY-XMP computer. The following sections outline each of the major components of the solution algorithm in more detail.

3.3 Scaling to the Structural Constraints

The structural constraints will generally be confined to constraints on static member stresses or nodal displacements for specified loading conditions, and/or to constraints on the open-loop structural frequencies. Scaling the entire structure to any one of these constraints is generally rather straightforward for the finite elements considered in this work.

The design variables for a truss/membrane/shear element type structure are the elemental cross-sectional areas/thicknesses. Then for a statically determinate structure, the design variables are inversely proportional to the static displacements and stresses. For indeterminate structures, this inverse proportionality is not exact, but is still a good approximation. In these cases, it will be necessary to perform the inverse scaling in an iterative procedure, but only a few iterations may be required to achieve the exact scaling desired, depending on the accuracy required.

Equal scaling of all the structural design variables will only affect the open-loop frequencies if the structure includes some nonstructural mass. Consider the Rayleigh quotient expression for an open-loop frequency $\lambda_i = \omega_i^2$ for a conservative, non-gyroscopic system, as

$$\lambda_i = \frac{\phi_i^T K_s \phi_i}{\phi_i^T [M_s + M_\ell] \phi_i} \tag{3.1}$$

where ϕ_i is the mode shape associated with λ_i, and M_s and M_ℓ are the mass matrices associated with the structural and nonstructural mass respectively. If the stiffness and structural mass matrices are linear functions of the design variables (true for truss, membrane and shear elements), then to obtain the *desired* open-loop frequency λ_i^d, consider scaling all design variables by the same scale factor s_i, to get

$$\lambda_i^d = \frac{s_i \phi_i^T K_s \phi_i}{\phi_i^T [s_i M_s + M_\ell] \phi_i} \tag{3.2}$$

Note that the nonstructural mass matrix is not scaled. If the ratio of actual open-loop frequency to desired open-loop frequency is

$$h_i = \frac{\lambda_i}{\lambda_i^d} \tag{3.3}$$

the scaling s_i necessary to obtain this ratio can be obtained by substituting equations (3.1) and (3.2) into equation (3.3), and solving for the scaling parameter s_i, to get

$$s_i = \frac{\phi_i^T M_\ell \phi_i}{\phi_i^T [h_i(M_s + M_\ell) - M_s] \phi_i} \tag{3.4}$$

One scaling parameter s_i for each frequency constraint will be obtained, and the scaling parameter of the most violated constraint would be used to scale the system.

3.4 Forming the Approximate Problem

Since the objective is linear in the design variables (when restricted to truss, membrane and/or shear finite elements), the objective gradient is constant over the entire design process and the objective function can be evaluated exactly for any intermediate design. Therefore the objective function need not be approximated at all. Approximation functions will be formed for both the structural and control constraint functions. The two most common types of approximation in use are the inverse design variable approximation (see [Can] for example) and the hybrid design variable approximation [Sta], which are both first-order Taylor's series expansions in some convenient intermediate design variable.

3.4.1 Inverse Design Variable Approximation

In this approximation the constraint function is expanded as a first-order Taylor's series expansion in variables that are the inverse of the design variables. As noted in a previous section, this approximation is exact for static structural constraints on a determinate structure, and this is the motivation for such a choice of intermediate variables for the structural design variables. Numerical experience gained in the current work has shown that this approximation also works quite well on the controller constraints in some cases.

A generic constraint approximation function written in terms of the intermediate design variables is then

$$\bar{g}_i(\boldsymbol{y}) = g_i(\boldsymbol{y}_0) + \sum_{j=1}^{N} \left.\frac{\partial g_i}{\partial y_j}\right|_{\boldsymbol{y}_0} (y_j - y_{j_0}) \tag{3.5}$$

where the $y_j = \frac{1}{p_j}$ are the (intermediate) inverse design variables. The constraint gradients are obtained with respect to the direct design variables $\boldsymbol{p}$, hence equation (3.5), written in terms of the p_j, becomes

$$\bar{g}_i(\boldsymbol{p}) = g_i(\boldsymbol{p}_0) - \sum_{j=1}^{N} p_{j_0}^2 \left.\frac{\partial g_i}{\partial p_j}\right|_{\boldsymbol{p}_0} \left(\frac{1}{p_j} - \frac{1}{p_{j_0}}\right) \tag{3.6}$$

Note that these approximation functions are nonlinear in the design variables p_j.

The gradients of these approximation functions can be found by differentiating equation (3.6) with respect to the direct design variables p_j, to get

$$\frac{\partial \bar{g}_i(\boldsymbol{p})}{\partial p_j} = \left.\frac{\partial g_i}{\partial p_j}\right|_{\boldsymbol{p}_0} \left(\frac{p_{j0}}{p_j}\right)^2 \tag{3.7}$$

Note that this gradient is not constant, but depends on the current design variable.

3.4.2 Hybrid Design Variable Approximation

The use of approximation functions is motivated by the desire to obtain a low-order approximation that best approximates the actual constraint behaviour and that leads to the least constraint violation possible. The inverse design variable formulation introduced above is a simple, low-order approximation that does a good job of approximating the constraint functions, especially for the structural constraints. However, it is not the most conservative approximation that could be obtained using first-order expansions, where the "most conservative" means the most positive for constraints of the type $g(\boldsymbol{p}) \leq 0$ and the most negative for constraints of the $g(\boldsymbol{p}) \geq 0$.

The hybrid design variable approximation obtains the most conservative possible approximation by combining the features of the inverse design variable approximation as presented above, and of a direct design variable approximation, where the expansion is performed with respect to the direct design variables [Sta]. Consider approximating the constraint function using the first two terms in the Taylor's series expansion that is linear in the direct design variables, to get

$$\bar{g}_L(\boldsymbol{p}) = g(\boldsymbol{p}_0) + \sum_{j=1}^{N} \left.\frac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} (p_j - p_{j0}) \tag{3.8}$$

or by an expansion that is linear in the inverse design variables y_j, as defined in the previous section, as

$$\bar{g}_I(\boldsymbol{y}) = g(\boldsymbol{y}_0) + \sum_{j=1}^{N} \left.\frac{\partial g}{\partial y_j}\right|_{\boldsymbol{y}_0} (y_j - y_{j0}) \tag{3.9}$$

Since our gradient information is with respect to the direct design variables p_j, $\bar{g}_I$

must be written as

$$\bar{g}_I(\boldsymbol{p}) = g(\boldsymbol{p}_0) - \sum_{j=1}^{N} p_{j0}^2 \frac{\partial g_i}{\partial p_j}\bigg|_{\boldsymbol{p}_0} \left(\frac{1}{p_j} - \frac{1}{p_{j0}}\right) \tag{3.10}$$

A criterion for determining the most conservative approximation from equation (3.8) and (3.10) is obtained [Sta] by subtracting equation (3.10) from (3.8), to get

$$\bar{g}_I(\boldsymbol{p}) - \bar{g}_L(\boldsymbol{p}) = -\sum_{j=1}^{N} \frac{1}{p_j} \frac{\partial g}{\partial p_j}\bigg|_{\boldsymbol{p}_0} (p_j - p_{j0})^2 \tag{3.11}$$

Since the constraint equations used in the problem formulation are expressed as $g(\boldsymbol{x}) \leq 0$, it can be seen, with reference to equation (3.11), that $\bar{g}_I$ is more conservative (greater) than $\bar{g}_L$ when

$$\frac{1}{p_j} \frac{\partial g}{\partial p_j}\bigg|_{\boldsymbol{p}_0} < 0 \tag{3.12}$$

and $\bar{g}_L$ is more conservative (greater) than $\bar{g}_I$ when

$$\frac{1}{p_j} \frac{\partial g}{\partial p_j}\bigg|_{\boldsymbol{p}_0} > 0 \tag{3.13}$$

The hybrid approximation formulation is motivated by the desire to obtain an approximation that best predicts the actual function behaviour without violating the constraint. Then let the hybrid approximation be given by

$$\bar{g}_H(\boldsymbol{p}) = g(\boldsymbol{p}) + \sum_{j=1}^{N} \mathcal{A}_j \frac{\partial g}{\partial p_j}\bigg|_{\boldsymbol{p}_0} \tag{3.14}$$

where

$$\mathcal{A}_j = \begin{cases} (p_j - p_{j_0}) & \text{if } \left.\dfrac{1}{p_j}\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} > 0 \\ \left(\dfrac{p_{j_0}}{p_j}\right)(p_j - p_{j_0}) & \text{if } \left.\dfrac{1}{p_j}\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} < 0 \end{cases} \tag{3.15}$$

With this hybrid approximation, a given constraint function may have a linear approximation with respect to one variable, and an inverse approximation with respect to another variable, hence the term "hybrid". If the p_j are physical design variables such as areas, they are restricted to be positive, and the term $1/p_j$ can be discarded from the conditions in equations (3.15).

If some of the p_j are non-physical design variables which are unrestricted in sign, they could become close to or equal to zero at some stage in the design process. In such a situation, an inverse design variables approximation becomes non-sensical, and the direct design variables approximation should be used. Accordingly, the general hybrid approximation definition for $\mathcal{A}_j$ in equation (3.15) should be modified to

$$\mathcal{A}_j = \begin{cases} (p_j - p_{j_0}) & \text{if } \left.\dfrac{1}{p_j}\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} > 0 \quad \text{or} \quad |p_j| < \Delta \\ \left(\dfrac{p_{j_0}}{p_j}\right)(p_j - p_{j_0}) & \text{if } \left.\dfrac{1}{p_j}\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} < 0 \end{cases} \tag{3.16}$$

where Δ is a "small' number specified by the user.

The gradients of the hybrid approximation function can be found by differentiating equation (3.14) with respect to the direct design variables p_j, to get

$$\frac{\partial \bar{g}_H(\boldsymbol{p})}{\partial p_j} = \begin{cases} \left.\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} & \text{if } \left.\dfrac{1}{p_j}\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} > 0 \quad \text{or} \quad |p_j| < \Delta \\ \left(\dfrac{p_{j_0}}{p_j}\right)^2 \left.\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} & \text{if } \left.\dfrac{1}{p_j}\dfrac{\partial g}{\partial p_j}\right|_{\boldsymbol{p}_0} < 0 \end{cases} \tag{3.17}$$

3.5 Move-limits

Move-limits are imposed on the design variables during each approximate problem solution. This is done in an attempt to restrain the design variables to a region in which the explicit function approximations remain reasonably accurate. However, deciding how to impose these move-limits is a non-trivial task. The local curvature of the design space (i.e. how nonlinear ar: the actual constraint surfaces in the region about the expansion point of the approximations) will determine the move-limits, with more strict move-limits applied in regions of high curvature, and less strict move-limits imposed in regions of low curvature. Since second-derivative information is required to estimate curvatures, and since such evaluations are very expensive computationally, imposing move-limits is usually reduced to an art based on past experience. It is possible to obtain the second derivatives using only first derivative information, and optimization algorithms that do this are termed quasi-Newton methods. However, these methods typically take N iterations to fill the Hessian, and can be very costly if N is large.

For the purpose of this work, a move-limits factor γ is imposed in an exponential form. If the current design variable and approximation expansion vector is $\boldsymbol{p}_0$, then the upper and lower bounds on the design variables for the current approximate problem are defined as

$$\begin{aligned} \boldsymbol{p}_u &= \gamma \boldsymbol{p}_0 \\ \boldsymbol{p}_\ell &= \frac{1}{\gamma} \boldsymbol{p}_0 \end{aligned} \tag{3.18}$$

where $\gamma \geq 1$. The limits specified in equation (3.18) must be imposed element by element. Note that since the design variables in this example will be structural design variables only, they are restricted to be positive. Obviously, equation (3.18) must be modified if the design variables can be negative. The exponential form of the move-limit factor is defined by the particular choice of γ_{min} and γ_{max}, as shown in Figure 3.2. The move limits get more restrictive as the optimization progresses, and as the design supposedly gets closer to its optimum point. Since the initial design may be very far from the optimum point, allowing large moves in the first few iterations enables the design to jump around the design space somewhat during the initial stages of the design process.

Initially, with γ large, the design is more likely to be trapped in regions of a deep local minima "well", and is more likely to escape from shallow local minima wells. As the move-limits become more restrictive, i.e. as $\gamma \rightarrow 1$, the design will become trapped in a local minimum. This process is not unlike the process of "simulated annealing" [Kir-2], althoug.. obviously not as general or controllable. Unfortunately, there are no general conditions which will indicate which local minimum is the global minimum, or where it can be found, except in cases where the design space is convex. The design space for this problem is definitely not convex.

3.6 Scaling to the Controller Constraints in the case of Full State Feedback Control

In the case when full state feedback is to be used, it was seen in Chapter 2 that it is reasonable to assume that the controller at each step in the design process is an appropriate LQR controller. In this case then, it becomes practical to scale the structure to the closest control effort constraint and closest output response constraint simultaneously. The variables with which the structure is scaled are the structural design variables (elemental areas or thicknesses), and the Lagrange multipliers associated with the two controller constraints λ_u and λ_y (where for clarity, and without loss of generality, the subscripts on the λ's that refer to the particular control effort or output response constraints under consideration have been dropped).

Note that changing the values of λ_u and λ_y cannot independently change the values of $u_{ms} = \mathrm{tr}(K^T RKX)$ and $y_{ms} = \mathrm{tr}(H_d^T W H_d X)$, because in the LQR problem, only the ratio of λ_u to λ_y is important. One can choose the ratio (λ_u/λ_y) to satisfy one of the control constraints — say u_{ms}. Then y_{ms} will not in general be satisfied. Suppose y_{ms} is too large (i.e. $y_{ms} > \alpha^2$) at the particular point where u_{ms} is satisfied. Then the *only* way one can satisfy the y_{ms} constraint is to increase the sizes of at least some of the structural members.

It makes sense that one needs to change the structure to satisfy both control constraints. If the control constraints could be satisfied by simply choosing appropriate controller parameters, then there would be no interaction between structural optimization and controller optimization. Intuitively, it can be seen that this is not the case. In the remainder of this section, a method is developed to scale both the ratio (λ_u/λ_y), and the cross-sectional areas of truss members, to exactly satisfy both mean square control constraints.

Note that each member of the structure will be scaled by the same amount to fulfill our goals. Obviously, this method is not absolutely mandated, and some other approach could be used where the design variables are not scaled equally. However, this would then be *resizing* rather than *scaling*, a process normally left to the nonlinear programming algorithm. Assume that, at iteration i, values for $(u_{ms})_i$, $(y_{ms})_i$, the $(p_j)_i$ for $j = 1, \ldots,$(number of structural members), and $(\lambda_u)_i$ (or equivalently, the ratio $(\lambda_u/\lambda_y)_i$), are known. Now define

$$\frac{(\lambda_u)_i}{(\lambda_u)_{i-1}} = \Delta_i \tag{3.19}$$

$$\frac{(p_j)_i}{(p_j)_{i-1}} = \delta_i \tag{3.20}$$

$$\frac{(u_{ms})_i}{(u_{ms})_i} = u_i \tag{3.21}$$

$$\frac{(y_{ms})_i}{(y_{ms})_{i-1}} = y_i \tag{3.22}$$

Our final scaling aim is to set $u_{ms} = \beta^2$ and $y_{ms} = \alpha^2$. Therefore, let

$$\frac{(u_{ms})_i}{\beta^2} = u_{i,\beta} \tag{3.23}$$

$$\frac{(y_{ms})_i}{\alpha^2} = y_{i,\alpha} \tag{3.24}$$

Further, it is assumed that the mean square values $(u_{ms})_i$ and $(y_{ms})_i$ will change, as a result of changes to $(\lambda_u)_i$ and the $(p_j)_i$, according to the equations

$$u_{i+1} = \frac{(u_{ms})_{i+1}}{(u_{ms})_i} = \Delta_{i+1}^{a_i}\delta_{i+1}^{b_i} \tag{3.25}$$

$$y_{i+1} = \frac{(y_{ms})_{i+1}}{(y_{ms})_i} = \Delta_{i+1}^{c_i}\delta_{i+1}^{d_i} \tag{3.26}$$

where a_i, b_i, c_i and d_i are constants.

One would first use initial (educated) guesses for a_i, b_i, c_i and d_i to find the appropriate changes to $(\lambda_u)_0$ and the $(p_j)_0$ required to satisfy $u_1 = \beta^2$ and $y_1 = \alpha^2$. The actual changes to u_0 and y_0 due to the initial values of a_0, b_0, c_0 and d_0 will probably not be those predicted, and the process will have to be repeated in an iterative fashion to obtain the desired result. However, by using the actual changes to u_i and y_i at each step, updated and improved values for the constants a_i, b_i, c_i and d_i can be estimated from the previous calculations. There are two steps to each iteration.

(a) Evaluating Δ_{i+1} and δ_{i+1} given a_i, b_i, c_i and d_i

We desire, at the next step, $(u_{ms})_{i+1} = \beta^2$ and $(y_{ms})_{i+1} = \alpha^2$. That is,

$$u_{i+1} = \frac{(u_{ms})_{i+1}}{(u_{ms})_i} = \Delta_{i+1}^{a_i}\delta_{i+1}^{b_i} = u_{i+1,\beta}^{-1} \tag{3.27}$$

$$y_{i+1} = \frac{(y_{ms})_{i+1}}{(y_{ms})_i} = \Delta_{i+1}^{c_i}\delta_{i+1}^{d_i} = y_{i+1,\alpha}^{-1} \tag{3.28}$$

(3.29)

These provide 2 equations in the 2 unknowns Δ_{i+1} and δ_{i+1}. The two unknowns can be found by taking logarithms of equations (3.27) and (3.28), to get

$$\log(\Delta_{i+1}) = -\frac{d_i \log(u_{i+1,\beta}) - b_i \log(y_{i+1,\alpha})}{a_i d_i - b_i c_i} \tag{3.30}$$

$$\log(\delta_{i+1}) = -\frac{a_i \log(y_{i+1,\alpha}) - c_i \log(u_{i+1,\beta})}{a_i d_i - b_i c_i} \tag{3.31}$$

which will be valid as long as

$$a_i d_i - b_i c_i \neq 0 \tag{3.32}$$

If this condition is violated, this would indicate that the proposed changes cannot independently scale u_{ms} and y_{ms}. However, it must be remembered that the constants a_i, b_i, c_i and d_i used here are not exactly known, so the condition (3.32) may sometimes be violated because of the particular values of the constants at that time. Numerical experience with this solution algorithm indicates that condition (3.32) is not restictive. If this condition does occur at some point, then the unknowns are set to their values on the previous iteration, and the determinant is unlikely to be zero again on the next iteration.

(b) Evaluating a_{i+1}, b_{i+1}, c_{i+1} and d_{i+1} given Δ_{i+1}, δ_{i+1}, u_{i+1} and y_{i+1}

The only restriction on the initial values a_0, b_0, c_0 and d_0 is that as defined by equation (3.32). However, one can obtain initial educated guesses by performing parameter tests on the system if one so desires. After the first iteration, there is only one back point to use, hence only two of the four parameters can be updated. In this work, the parameters updated are a_0 and d_0, while setting $b_1 = b_0$ and $c_1 = c_0$. We know Δ_1, δ_1, u_1 and y_i, and wish to fit a_1 and d_1 to

$$u_1 = \Delta_1^{a_1} \delta_1^{b_0} \tag{3.33}$$

$$y_1 = \Delta_1^{c_0} \delta_1^{d_1} \tag{3.34}$$

Again, taking logarithms of equations (3.33) and (3.34) gives

$$a_1 = \frac{\log(u_1) - b_0 \log(\delta_1)}{\log(\Delta_1)} \tag{3.35}$$

$$d_1 = \frac{\log(y_1) - c_0 \log(\Delta_1)}{\log(\delta_1)} \tag{3.36}$$

After the second iteration there are back points available, so all of the parameters can be updated. Consider

$$u_{i+1} = \Delta_{i+1}^{a_{i+1}} \delta_{i+1}^{b_{i+1}} \tag{3.37}$$

$$y_{i+1} = \Delta_{i+1}^{c_{i+1}} \delta_{i+1}^{d_{i+1}} \tag{3.38}$$

$$u_i = \Delta_i^{a_{i+1}} \delta_i^{b_{i+1}} \tag{3.39}$$

$$y_i = \Delta_i^{c_{i+1}} \delta_i^{d_{i+1}} \tag{3.40}$$

$$\tag{3.41}$$

and solving equations (3.37) through (3.40) for the unknowns gives

$$a_{i+1} = \frac{\log(\delta_i) \log(u_{i+1}) - \log(\delta_{i+1}) \log(u_i)}{\log(\Delta_{i+1}) \log(\delta_i) - \log(\Delta_i) \log(\delta_{i+1})} \tag{3.42}$$

$$b_{i+1} = \frac{\log(\Delta_{i+1}) \log(u_i) - \log(\Delta_i) \log(u_{i+1})}{\log(\Delta_{i+1}) \log(\delta_i) - \log(\Delta_i) \log(\delta_{i+1})} \tag{3.43}$$

$$c_{i+1} = \frac{\log(\delta_i) \log(y_{i+1}) - \log(\delta_{i+1}) \log(y_i)}{\log(\Delta_{i+1}) \log(\delta_i) - \log(\Delta_i) \log(\delta_{i+1})} \tag{3.44}$$

$$d_{i+1} = \frac{\log(\Delta_{i+1}) \log(y_i) - \log(\Delta_i) \log(y_{i+1})}{\log(\Delta_{i+1}) \log(\delta_i) - \log(\Delta_i) \log(\delta_{i+1})} \tag{3.45}$$

which will be valid as long as

$$\log(\Delta_{i+1})\log(\delta_i) - \log(\Delta_i)\log(\delta_{i+1}) \neq 0 \tag{3.46}$$

Once again, if this condition is violated, then the unknowns should be set to their values on the previous iteration, and the iteration repeated.

3.7 Structural Analysis and State Space Formulation

Large flexible structures are distributed parameter systems; that is, their mass, damping and stiffness characteristics are described by variables depending on time and space. Therefore, the governing equations of motion will be partial differential equations, which are theoretically of infinite dimension. To model these structures in a way which facilitates easy and efficient solutions to the equations of motion, the structures are normally discretized, commonly by using the well known finite element technique (NASTRAN for example). Discretization reduces the structure to a finite degree of freedom system, where the governing equations of motion will be ordinary differential equations of finite dimension.

In this work, a finite element model of the structure is formed, and only structures which are a collection of truss elements are considered. The problem formulation and solution techniques employed are general however, and the extension to structures composed of beam or plate elements should be straightforward. A truss element allows only axial forces and displacements, and for a fixed material and geometry, the only design parameter is the cross- sectional area. Other elements may exhibit more design freedom. For example, a beam element may have cross-sectional area and area moments of inertia as design parameters.

The equations of motion for the discretized structure, subject to control forces in a disturbance environment, can be written in the form

$$M\ddot{\boldsymbol{\xi}} + C\dot{\boldsymbol{\xi}} + K_s\boldsymbol{\xi} = D\boldsymbol{u} + G_{w_1}\boldsymbol{w} \tag{3.47}$$

where $\boldsymbol{\xi}$ is a q-vector of generalized displacements (q is the number of degrees of freedom in the structure), $\boldsymbol{u}$ is an n_u-vector of applied control forces, $\boldsymbol{w}$ is an n_w-vector of disturbance forces, M is the system mass matrix, K_s is the system stiffness matrix, C is a damping matrix, D is a matrix describing the locations and orientations of the force actuators, and G_{w_1} is a matrix describing the points of application and orientation of the disturbances.

The system mass and stiffness matrices M and K_s are built up from information about the elemental material properties, geometry, location, orientation, and boundary conditions. Knowledge of M and K_s enables calculation of such open-loop structural information as element stresses, nodal displacements, and open-loop natural frequencies. The damping matrix C can be formed during the finite element process using special damping elements. If this option is not desirable or available, the damping effect can be included in a variety of ways (see Section 3.7.3).

The mass and stiffness matrices will be functions of the free elemental parameters. Furthermore, these matrices will be linear functions of these structural design variables for structures comprised of only truss, membrane and/or shear elements. For the case considered in this work, where our structures are composed of only truss elements, the structural design variables will be the cross-sectional area of each element.

Once the finite element analysis has been performed, and the mass and stiffness (and possibly damping) matrices obtained, the state space system can be formed. This involves writing the second order equations of motion (3.47) as a first-order state space system of equations, in which a choice of "state" must be made. There are many such states that could be chosen, but two choices are the most common.

3.7.1 Physical State Space Model

A state space system based on physical variables can be formed by defining the $n = 2q$-dimensional state vector $\boldsymbol{x}$ as

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\xi} \\ \dot{\boldsymbol{\xi}} \end{bmatrix} \tag{3.48}$$

The elements of this $\boldsymbol{x}$ are the actual degrees of freedom of the elements. Then, the state space matrices F, G, and G_w of the realization of equations (2.1) can be formed as

$$F = \begin{bmatrix} 0 & I \\ -M^{-1}K_s & -M^{-1}C \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ M^{-1}D \end{bmatrix}, \quad G_w = \begin{bmatrix} 0 \\ M^{-1}G_{w_1} \end{bmatrix} \tag{3.49}$$

A general design output matrix H_d will be of the form

$$H_d = \begin{bmatrix} H_1 & H_2 \end{bmatrix} \tag{3.50}$$

where H_1 gives the contribution to this design goal from the displacement states, and H_2 gives the contribution to this design goal from the velocity states.

3.7.2 Modal State Space Model

An alternative choice for state space representation is to first form the equations of motion (3.47) in modal form [Mei], as

$$\ddot{\boldsymbol{\eta}} + \Phi^T C \Phi \dot{\boldsymbol{\eta}} + [\omega_i^2]\boldsymbol{\eta} = \Phi^T D \boldsymbol{u} + \Phi^T G_{w_1} \boldsymbol{w} \tag{3.51}$$

where $\boldsymbol{\eta}$ is the vector of generalized modal displacements such that $\boldsymbol{\xi} = \Phi\boldsymbol{\eta}$, $[\omega_i^2]$ is a diagonal matrix of natural frequencies squared, and Φ is the matrix whose columns are the mode shapes corresponding to the natural modes of vibration, which are assumed to be normalized such that $\Phi^T M \Phi = I$.

To form the state space realization in terms of these *modal* variables, define the state vector $\boldsymbol{x}$ as

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\eta} \\ \dot{\boldsymbol{\eta}} \end{bmatrix} \tag{3.52}$$

The elements of this $\boldsymbol{x}$ are the generalized displacements and velocities of the modal coordinates $\boldsymbol{\eta}$. Then, the state space matrices F, G, and G_w of the realization of equations (2.1) can be formed as

$$F = \begin{bmatrix} 0 & I \\ -[\omega_i^2] & -\Phi^T C \Phi \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ \Phi^T D \end{bmatrix}, \quad G_w = \begin{bmatrix} 0 \\ \Phi^T G_{w_1} \end{bmatrix} \tag{3.53}$$

Note that for a general C generated from passive damping element information (if this is even possible), the transformed damping matrix $\Phi^T C \Phi$ will generally not be diagonal.

A general design output matrix H_d will be given by

$$H_d = \begin{bmatrix} H_1 \Phi & H_2 \Phi \end{bmatrix} \tag{3.54}$$

3.7.3 Damping Models

Damping of the structure is difficult to model in the traditional finite element sense. Normally, damping in a certain form is assumed after the formation of the mass and stiffness matrices. One form is Rayleigh damping [Bat], where the damping matrix is formed as a linear combination of the mass and stiffness matrices as

$$C = a_0 M + a_1 K_s \tag{3.55}$$

If the damping ratios of some or all modes are either known or to be specified, C can be defined using the Caughey series [Bat]

$$C = M \sum_{k=0}^{p-1} a_k [M^{-1} K]^k \tag{3.56}$$

where $p \leq n$ damping ratios $\{\zeta_i,\ i = 1, \ldots, p\}$ are specified, and the coefficients $\{a_k,\ k = 1, \ldots, p\}$ are calculated from the p simultaneous equations

$$\zeta_i = \frac{1}{2}\left(\frac{a_0}{\omega_i} + a_1\omega_i + a_2\omega_i^3 + \cdots + a_{p-1}\omega^{2p-3}\right) \qquad \text{for } i = 1, \ldots, p \tag{3.57}$$

Note that with $p = 2$, this result reduces to the Rayleigh damping as in equation (3.55). For large p in equations (3.57), the computational effort involved in finding the coefficients a_k becomes quite large. Also, assuming C must be differentiated at some stage, the Rayleigh damping as in equation (3.55) or the proportional damping as in equations (3.61) is in a much better form than that of equation (3.57). A disadvantage of Rayleigh damping is that the higher modes tend to be more highly damped than conventional wisdom dictates.

A damping model where the damping ratio in each mode is the same, and where the modal damping itself is proportional to the modal frequency, can be formed by defining C as

$$C = 2\zeta K^{1/2} \tag{3.58}$$

where $K^{1/2}$ is the symmetric square root of K such that

$$K^{1/2}K^{1/2} = K \tag{3.59}$$

and ζ is the damping ratio of the modes. Note that with the C as defined by equation (3.58), the transformed damping matrix in a modal state space representation will be

$$\begin{aligned} \Phi^T C \Phi &= 2\zeta\Phi^T K^{1/2}\Phi \\ &= 2\zeta[\omega_i] \\ &= [2\zeta\omega_i] \end{aligned} \tag{3.60}$$

where $[\omega_i]$ is a diagonal matrix of modal frequencies, and $[2\zeta\omega_i]$ is a diagonal damping matrix. This method of forming a damping matrix is also probably not practical due to the effort involved. The diagonal nature of $[2\zeta\omega_i]$ means that the modal equations of motion (3.51) will decouple with this form of damping into equations of motion for each mode. However, each mode is restricted with this form of damping to have the same damping ratio and can be formed by simply *specifying* $\Phi^T C \Phi$ to be a diagonal matrix $[2\zeta_i\omega_i]$. That is, assume proportional damping of the form

$$\phi_i^T C \phi_j = 2\zeta_i\omega_i\delta_{ij} \tag{3.61}$$

where ζ_i is the *modal* damping parameter, and δ_{ij} is the Kronecka delta. The assumption in proportional damping is that the total damping in the structure is the sum of the individual damping in each mode.

For state space realizations based on physical variables in this work, Rayleigh damping as in equation (3.55) was used to model the structural damping. For state space realizations based on modal variables in this work, damping of the form of equation (3.61) was used to model the damping. The particular values for the constants a_0, a_1, and ζ_i used are given later.

3.8 Gradient Analysis

For the numerical optimization procedure to be practical, especially as the dimensionality increases to realistic structures, it is essential that it be possible to evaluate the first-order sensitivities of the complex objective and constraint equations in an efficient manner. The following sections summarize the analytical derivation of the appropriate gradients, where it will be seen that such efficient expressions can indeed be found.

3.8.1 Objective Function Gradient

For structures formed from truss/membrane/plate type elements, the objective function (the weight) is a linear function of the finite element thicknesses and/or cross sectional areas. That is,

$$J(\boldsymbol{p}^s) = \sum_{i=1}^{N_s} \rho_i L_i p_i^s + \sum_{j=1}^{N_\ell} m_j \tag{3.62}$$

where N_s is the number of structural design variables (number of finite elements, dimension of $\boldsymbol{p}^s$), ρ_i, L_i and p_i are the density, fixed geometric size (length of a truss element, or area of a membrane or shear element) and the structural design variable (cross-sectional area for a truss element, or thickness of a membrane or shear element) of the i^{th} finite element, N_ℓ is the number of nonstructural lumped masses on the structure, and m_j is the j^{th} nonstructural lumped mass. The gradient of the objective function is then easily constructed as

$$\frac{\partial J(\boldsymbol{p}^s)}{\partial p_i^s} = \rho_i L_i \qquad \text{for } i = 1, \ldots, N_s \tag{3.63}$$

which is constant for all designs. Note that the gradient of the objective function with respect to any *controller* design variable will be zero, since in this work the mass of the controllers was not considered.

3.8.2 Controller Constraint Gradients for the case of Full State Feedback Control

The controller constraints are given by

$$g_{cc_i} = \frac{\mathrm{tr}[K_i^T R_i K_i X]}{\beta_i^2} - 1 \leq 0 \qquad \text{for } i = 1, \ldots, n_\beta \tag{3.64}$$

$$g_{oc_i} = \frac{\mathrm{tr}[H_{d_i}^T W_i H_{d_i} X]}{\alpha_i^2} - 1 \leq 0 \qquad \text{for } i = 1, \ldots, n_\alpha \tag{3.65}$$

where X is the covariance matrix, found as the solution to the Lyapunov equation

$$F_{cl} X + X F_{cl}^T + G_w X_w G_w^T = 0 \tag{3.66}$$

where

$$F_{cl} = (F - GK) \tag{3.67}$$

For this particular case, it is assumed that the feedback gain matrix K at any point is given by the solution of the appropriate LQR problem as

$$K = R^{-1}G^T\Lambda_x \tag{3.68}$$

$$F^T\Lambda_x + \Lambda_x F - \Lambda_x G R^{-1} G^T \Lambda_x + W = 0 \tag{3.69}$$

where

$$R = \operatorname{diag}\left\{\left(\frac{\lambda_{u_i}}{\beta_i^2}\right) R_i\right\} \tag{3.70}$$

$$W = \sum_{i=1}^{n_\alpha} \left(\frac{\lambda_{y_i}}{\alpha_i^2}\right) H_{d_i}^T W_i H_{d_i} \tag{3.71}$$

This assumption is based on the analysis that showed that the first-order Kuhn-Tucker optimality conditions for the controller variables were satisfied for an LQR controller of this type. However, this result only holds at the optimum point, and not necessarily everywhere within the design space. This introduces another constraint into the problem, namely the LQR constraint, similar to the covariance equation constraint. Using this approach then, both X *and* K are now treated as variables dependent on the particular values of the structural design variables $\boldsymbol{p}^s$.

Therefore, if scalar p_j represents a structural design variable, the controller constraint gradients are given by

$$\frac{\partial g_{cc_i}}{\partial p_j} = \frac{1}{\beta_i^2}\operatorname{tr}\left[\left(\frac{\partial K_i^T}{\partial p_j} R_i K_i + K_i^T R_i \frac{\partial K_i}{\partial p_j}\right) X + K_i^T R_i K_i \frac{\partial X}{\partial p_j}\right] \tag{3.72}$$

$$\frac{\partial g_{oc_i}}{\partial p_j} = \frac{1}{\alpha_i^2}\mathrm{tr}\left[\left(\frac{\partial H_{d_i}^T}{\partial p_j}W_i H_{d_i} + H_{d_i}^T W_i \frac{\partial H_{d_i}}{\partial p_j}\right) X + H_{d_i}^T W_i H_{d_i}\frac{\partial X}{\partial p_j}\right] \tag{3.73}$$

With K defined as in equation (3.68), its derivative with respect to p_j becomes

$$\frac{\partial K}{\partial p_j} = R^{-1}\left[\frac{\partial G^T}{\partial p_j}\Lambda_x + G^T\frac{\partial \Lambda_x}{\partial p_j}\right] \tag{3.74}$$

where $\frac{\partial \Lambda_x}{\partial p_j}$ can be found by differentiating equation (3.69), and using equation (3.68), to be

$$F_{cl}^T\frac{\partial \Lambda_x}{\partial p_j} + \frac{\partial \Lambda_x}{\partial p_j}F_{cl} = -\left[\left(\frac{\partial F}{\partial p_j} - \frac{\partial G}{\partial p_j}K\right)^T \Lambda_x + \Lambda_x\left(\frac{\partial F}{\partial p_j} - \frac{\partial G}{\partial p_j}K\right) + \frac{\partial W}{\partial p_j}\right] \tag{3.75}$$

where

$$\frac{\partial W}{\partial p_j} = \sum_{i=1}^{n_\alpha}\left(\frac{\lambda_{y_i}}{\alpha_i^2}\right)\left(\frac{\partial H_{d_i}^T}{\partial p_j}W_i H_{d_i} + H_{d_i}^T W_i\frac{\partial H_{d_i}}{\partial p_j}\right) \tag{3.76}$$

Note that only the right-hand side of the Lyapunov eqaution (3.75) changes for differing p_j. Since most of the effort involved in the numerical solution of Lyapunov equations is in either a decomposition phase (for direct solvers) or an inversion phase (iterative solvers) operating on the left-hand side of the equation, one decomposition or set of inversions will be enough to solve equation (3.75) for all p_j.

The partitions $\frac{\partial K_i}{\partial p_j}$ of $\frac{\partial K}{\partial p_j}$ required for each of the control effort constraint gradients as in equation (3.72) can be found easily knowing the partitions of K used to define the original constraint functions (3.64). From equations (3.72), the gradient of the covariance matrix is required, and can be found by differentiating equation (3.66) to get

$$F_{cl}\frac{\partial X}{\partial p_j} + \frac{\partial X}{\partial p_j}F_{cl}^T + \mathcal{H}_j = 0 \tag{3.77}$$

where

$$\mathcal{H}_j = \left[\frac{\partial F_{cl}}{\partial p_j}X + X\frac{\partial F_{cl}^T}{\partial p_j} + \frac{\partial G_w}{\partial p_j}X_w G_w^T + G_w X_w \frac{\partial G_w^T}{\partial p_j}\right] \tag{3.78}$$

$$\frac{\partial F_{cl}}{\partial p_j} = \left(\frac{\partial F}{\partial p_j} - \frac{\partial G}{\partial p_j}K - G\frac{\partial K}{\partial p_j}\right) \tag{3.79}$$

Note again that only the right-hand side of equation (3.77) changes for differing p_j. Further, equation (3.77) is the *adjoint* of equation (3.75), which means that only *one* decomposition or set of inverse calculations (which will of course need to be stored) of the left-hand side of equation (3.75) is required to solve *both* equations (3.75) and (3.77) for all p_j. This property significantly improves the efficiency of evaluating the gradients for this particular problem. However, the Lyapunov equation (3.77) must still be solved for every p_j. To avoid this, define the matrices $\mathcal{P}_i$ and $\mathcal{Q}_i$ as the symmetric positive definite solutions to the respective Lyapunov equations [Can]

$$F_{cl}^T\mathcal{P}_i + \mathcal{P}_i F_{cl} + K_i^T R_i K_i = 0 \tag{3.80}$$

$$F_{cl}^T\mathcal{Q}_i + \mathcal{Q}_i F_{cl} + H_{d_i}^T W_i H_{d_i} = 0 \tag{3.81}$$

so that

$$K_i^T R_i K_i = -\left(F_{cl}^T\mathcal{P}_i + \mathcal{P}_i F_{cl}\right) \tag{3.82}$$

$$H_{d_i}^T W_i H_{d_i} = -\left(F_{cl}^T\mathcal{Q}_i + \mathcal{Q}_i F_{cl}\right) \tag{3.83}$$

Substituting equations (3.82) and (3.83) into equations (3.72) gives

$$\frac{\partial g_{cc_i}}{\partial p_j} = \frac{1}{\beta_i^2}\mathrm{tr}\left[\left(\frac{\partial K_i^T}{\partial p_j}R_i K_i + K_i^T R_i \frac{\partial K_i}{\partial p_j}\right)X - \left(F_{cl}^T\mathcal{P}_i + \mathcal{P}_i F_{cl}\right)\frac{\partial X}{\partial p_j}\right]$$

$$\frac{\partial g_{oc_i}}{\partial p_j} = \frac{1}{\alpha_i^2}\mathrm{tr}\left[\left(\frac{\partial H_{d_i}^T}{\partial p_j}W_iH_{d_i} + H_{d_i}^TW_i\frac{\partial H_{d_i}}{\partial p_j}\right)X - \left(F_{cl}^T\mathcal{Q}_i + \mathcal{Q}_iF_{cl}\right)\frac{\partial X}{\partial p_j}\right] \tag{3.84}$$

Since tr$[AB]$ = tr$[BA]$ for any square AB, the above equations can be rewritten as

$$\frac{\partial g_{cc_i}}{\partial p_j} = \frac{1}{\beta_i^2}\mathrm{tr}\left[\left(\frac{\partial K_i^T}{\partial p_j}R_iK_i + K_i^TR_i\frac{\partial K_i}{\partial p_j}\right)X - \mathcal{P}_i\left(F_{cl}\frac{\partial X}{\partial p_j} + \frac{\partial X}{\partial p_j}F_{cl}^T\right)\right]$$

$$(3.85)$$

$$\frac{\partial g_{oc_i}}{\partial p_j} = \frac{1}{\alpha_i^2}\mathrm{tr}\left[\left(\frac{\partial H_{d_i}^T}{\partial p_j}W_iH_{d_i} + H_{d_i}^TW_i\frac{\partial H_{d_i}}{\partial p_j}\right)X - \mathcal{Q}_i\left(F_cl\frac{\partial X}{\partial p_j} + \frac{\partial X}{\partial p_j}F_{cl}^T\right)\right]$$

and with reference to equation (3.77) becomes

$$\frac{\partial g_{cc_i}}{\partial p_j} = \frac{1}{\beta_i^2}\mathrm{tr}\left[\left(\frac{\partial K_i^T}{\partial p_j}R_iK_i + K_i^TR_i\frac{\partial K_i}{\partial p_j}\right)X + \mathcal{P}_i\mathcal{H}_j\right]$$

$$(3.86)$$

$$\frac{\partial g_{oc_i}}{\partial p_j} = \frac{1}{\alpha_i^2}\mathrm{tr}\left[\left(\frac{\partial H_{d_i}^T}{\partial p_j}W_iH_{d_i} + H_{d_i}^TW_i\frac{\partial H_{d_i}}{\partial p_j}\right)X + \mathcal{Q}_i\mathcal{H}_j\right]$$

The advantage of equations (3.86) over equations (3.72) is that only the two Lyapunov equations (3.80) and (3.81), for $\mathcal{P}_i$ and $\mathcal{Q}_i$ respectively, need be solved to find the constraint gradients for every p_j, rather than the N Lyapunov equation solutions required to evaluate the covariance equation gradient for every p_j. Of course, $\mathcal{H}_j$ must still be evaluated for every p_j, but this would have been required anyway in equation (3.77).

3.8.3 State Space Matrix Gradients

The state space matrices are given by equations (3.49) and (3.50) for a realization based on physical variables, or equations (3.53) and (3.54) for a realization based on modal variables. For each of thses cases, the gradients of these state space matrices with respect to the structural design variables is required. Obviously, the gradients of these matrices with respect to the controller design variables will be zero.

Let p_j represent a structural design variable. Then for a realization based on physical variables, the gradients of the state space matrices are given by

$$\frac{\partial F}{\partial p_j} = \begin{bmatrix} 0 & 0 \\ \left(-\frac{\partial M^{-1}}{\partial p_j}K_s - M^{-1}\frac{\partial K_s}{\partial p_j}\right) & \left(-\frac{\partial M^{-1}}{\partial p_j}C - M^{-1}\frac{\partial C}{\partial p_j}\right) \end{bmatrix} \quad (3.87)$$

$$\frac{\partial G}{\partial p_j} = \begin{bmatrix} 0 \\ \frac{\partial M^{-1}}{\partial p_j}D \end{bmatrix} \quad (3.88)$$

$$\frac{\partial G_w}{\partial p_j} = \begin{bmatrix} 0 \\ \frac{\partial M^{-1}}{\partial p_j}G_{w_1} \end{bmatrix} \quad (3.89)$$

where

$$\frac{\partial M^{-1}}{\partial p_j} = -M^{-1}\frac{\partial M}{\partial p_j}M^{-1} \quad (3.90)$$

is the derivative of the inverse of the mass matrix M with respect to the structural design variable p_j. The derivative of M can be found very easily since it is simply the derivative of the local elemental mass matrix expressed in the global coordinate system. In the case when physical variables are used as above, the design output matrices H_{d_i} will not be functions of the structural design variables, so that

$$\frac{\partial H_{d_i}}{\partial p_j} = 0 \quad (3.91)$$

For a realization based on modal variables, the gradients of the state space matrices are given by

$$\frac{\partial F}{\partial p_j} = \begin{bmatrix} 0 & 0 \\ -A_1 & -A_2 \end{bmatrix} \tag{3.92}$$

$$\frac{\partial G}{\partial p_j} = \begin{bmatrix} 0 \\ \frac{\partial \Phi^T}{\partial p_j} D \end{bmatrix} \tag{3.93}$$

$$\frac{\partial G_w}{\partial p_j} = \begin{bmatrix} 0 \\ \frac{\partial \Phi^T}{\partial p_j} G_{w_1} \end{bmatrix} \tag{3.94}$$

where

$$A_1 = \text{diag}\left\{ \frac{\partial \lambda_1}{\partial p_j} \; \frac{\partial \lambda_2}{\partial p_j} \; \cdots \; \frac{\partial \lambda_n}{\partial p_j} \right\} \tag{3.95}$$

$$A_2 = \text{diag}\left\{ \frac{\zeta_1}{\sqrt{\lambda_1}} \frac{\partial \lambda_1}{\partial p_j} \; \frac{\zeta_2}{\sqrt{\lambda_2}} \frac{\partial \lambda_2}{\partial p_j} \; \cdots \; \frac{\zeta_n}{\sqrt{\lambda_n}} \frac{\partial \lambda_n}{\partial p_j} \right\} \tag{3.96}$$

The design output matrices H_{d_i} will now be functions of the structural design variables because they will be functions of the eigenvector matrix Φ. Therefore, the derivative of H_{d_i} is given by

$$\frac{\partial H_{d_i}}{\partial p_j} = \begin{bmatrix} H_1 \frac{\partial \Phi}{\partial p_j} & H_2 \frac{\partial \Phi}{\partial p_j} \end{bmatrix} \tag{3.97}$$

For the case when modal variables are used as above, the derivatives of the eigenvalues and eigenvectors will be required. This poses no problem if the instantaneous structure has no repeated eigenvalues, and the relevant expressions can be found, for example, in [Fox, Nel]. However, most researchers in the past seemed to avoid cases where repeated structural frequencies were present. Only very recently has this issue been addressed [Mil-2, Jua], and methods to evaluate the eigenvalue and eigenvector derivatives for the case where repeated eigenvalues are present are given in Appendix B.

3.9 Design Variable Linking

Even though the control design variables have been removed as active design variables in this case, the number of structural design variables can still be very large. An optimal solution where all structural elements have been treated as independent would probably define all elements as being a different size. In practical terms, it is not desirable to have a structure where *every* element is different, because this can cause production and spares problems. A process known as design variable linking can serve to reduce the number of different sized structural elements that will be needed. The simplest case of linking is to assign a single variable to a group of elements, so that all elements in that group will have the same variable value. Note that this process in effect introduces more constraints into the design process, which can affect the optimal solution found. However, a trade off between weight and ease of production may be necessary.

If there is no a priori knowledge of the optimal structure should be somehow symmetric, it is difficult to incorporate design variable linking into the solution procedure. Perhaps a better approach is to consider the linking only after initial results for unlinked designs have been evaluated. Regardless, design variable linking should be carefully considered at some stage in the design process for any practical design.

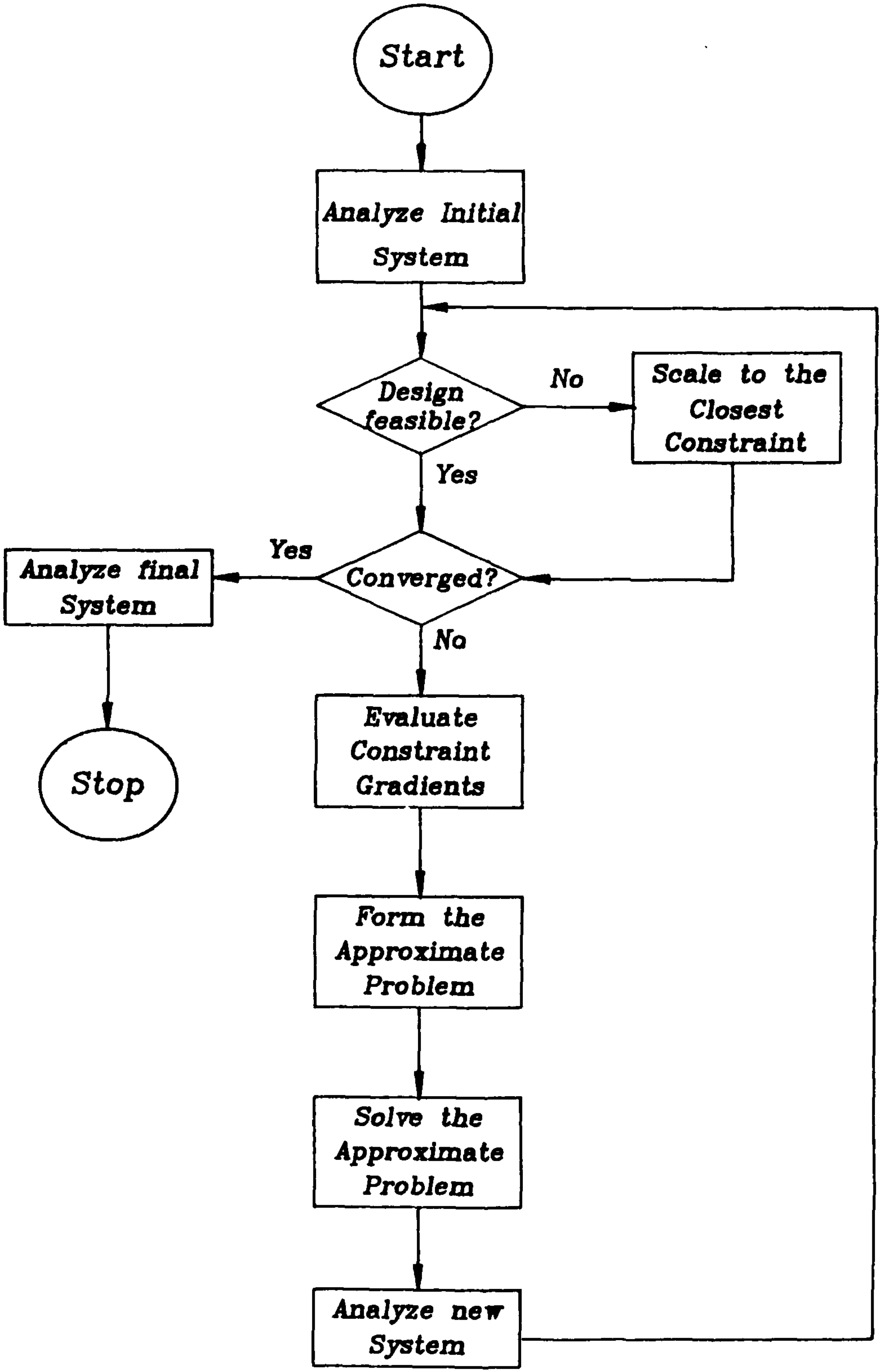
Start
Analyze Initial System
Design feasible?
No
Scale to the Closest Constraint
Yes
Yes
Analyze final System
Converged?
No
Stop
Evaluate Constraint Gradients
Form the Approximate Problem
Solve the Approximate Problem
Analyze new System

Figure 3.1: Algorithm for Solution by Mathematical Programming and Approximation Techniques

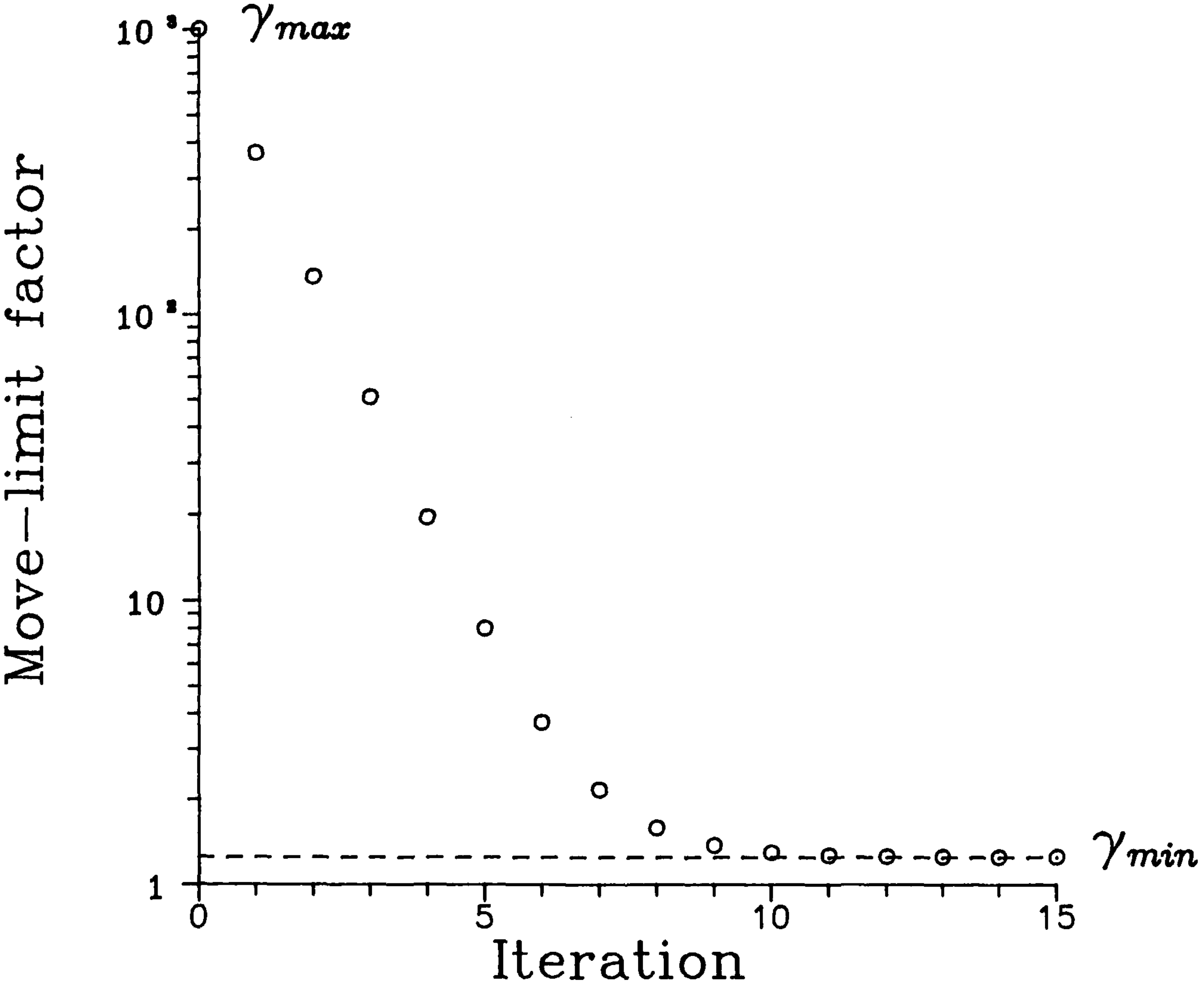

Figure 3.2: Move-limits for Mathematical Programming and Approximation Techniques

Chapter 4

Example: The DRAPER I Structure

4.1 The DRAPER I Structure

The DRAPER I structure [Str], which has been used as a generic flexible spacecraft model, is a tetrahedral truss attached to the ground by three right-angled bipods, as shown in Figure 4.1. Although attached to the ground, this model will act as a typical flexible structure pointing subsystem (e.g. antenna, radar, optical) attached to a rigid core. Any motion would then be with respect to this rigid core, and transmit forces to it. Consequently, this model has no rigid body degrees of freedom. The finite element model has 12 truss elements, since the joints are pinned and transmit no moments. There are four nodes that are free to move in all directions, so the model contains 12 degrees of freedom. The structural design variables are the cross-sectional areas of each of the 12 truss elements.

In [Str], the structure is set up as a non-dimensional model. However, for our purposes we use a dimensional model with material parameters of $\rho = 0.1\,\mathrm{lb/in}^3$, E = Young's Modulus = 20 kpsi. The dimensional values E and ρ were chosen to give initial numerical values of structural frequencies for the dimensional model roughly comparable to those of the non-dimensional model. The nodal coordinates are listed in Table 4.1, giving the length of the six upper bars as 10 feet and the length of the lower six bipod bars as $\sqrt{8}$ feet in this case. The structural members are numbered as defined by the finite element model connectivity data, summarized in Table 4.2. This dimensional model contains no non-structural mass.

Elements 7 through 12, the three right-angled bipods, take on the duties of velocity sensors and colocated force actuators. Only one output response constraint is defined ($n_\alpha = 1$), with the design output vector y_d representing the line-of-sight error of the top vertex [(x, y) displacements of vertex 1]. The disturbances, labelled w_1 and w_2 in Figure 4.2, are assumed to be independent, zero mean, Gaussian disturbances with intensity 1.0.

Since there are 12 degrees of freedom in the model for this structure, the state-space model will be 24^{th} order. There will be 6 inputs and 6 outputs, corresponding to the 6 legs of the structure. The damping added to the state space system will depend on the state space realization used. For cases where a realization based on physical variables is used, the damping matrix C is formed to be

$$C = 0.1M + 0.001K$$

For cases where a realization based on modal variables was used, the damping ratio of each mode was specified to be 0.1 % of the modal frequencies during the formation of the state matrices.

Only one control effort and one output response constraint are considered in this example. The weighting matrices R and W are set to the identity matrices, so that equal weighting is given to all components of $\boldsymbol{u}$ and $\boldsymbol{y}_d$. The minimum cross-sectional areas for all elements was specified as 0.1 in^2. For this problem, no static structural constraints were specified, the intent being to investigate the effect of the closed-loop controller constraints on the structural design optimization.

4.2 Numerical Considerations

Initially, nominal values of all elements are specified arbitrarily. For differing values of allowable expected output response (α^2) and allowable expected control effort (β^2), this initial design is scaled by adjusting the cross-sectional areas and Lagrange multiplier ratio, to meet the controller constraints. Using the gradient information, the design variables and the feedback gains are computed using the approximate formulations to further reduce the system mass. If, upon solution of the approximate problem, constraints are violated, the structure is scaled to the constraint surfaces using the scaling scheme discussed. This will change the weight from that value returned as the optimal approximate problem weight. Therefore, even though the optimal approximate problem weight will always be less than or equal to the weight before the approximate problem solution, after the scaling the weight change for that global iteration may sometimes be positive.

If the approximation functions are a high fidelity representation of the actual constraint surfaces, then during the solution of the optimization problem, we would not expect the solution to stray far from these actual constraint surfaces. That is, after each iteration, the actual constraint values would be close to the values predicted by the approximate functions In practice, this is only true when the approximate problem domain is restricted to a small region about the expansion point via the move-limits.

During the first few global iterations of the solution algorithm, the move-limits are on the order of γ_{max}, which is nominally set at 1000. This allows virtually unlimited movement in the design space. After several global iterations, the move-limits are reduced, in the manner described in Section 3.6, to the order of γ_{min}. This parameter can significantly affect the convergence properties of the solution algorithm, and is nominally set at 1.25, allowing the design variables to change by +25% and -20% in one iteration.

Two problems may occur when $\gamma \approx \gamma_{min}$. First, if the approximation functions predict the actual constraint behaviour very well, this is evidence that the constraint surface is highly linear over large region. In such a case, convergence may be very slow, and would be improved by increasing γ_{min}. The reverse may occur if in this limiting case the approximate functions predict the actual constraint behaviour very poorly. This is evidence of high local curvature of the constraint surfaces, and a reduction in the move limits by reducing γ_{min} will generally improve convergence. In fact, obtaining ultimate convergence may actually require such a reduction.

When obtaining optimal designs for differing β^2 and α^2 values, it is sometimes necessary to alter the value of γ_{min} to take advantage of local conditions near an optimum point. This can be accomplished by either manually stopping the solution process and changing γ_{min} before restarting, or by developing some adaptive online strategy to accomplish the same task. Changing the objectives of the optimization, by altering α^2 and β^2 for example, will often require a change in γ_{min}. The design process seems to be insensitive to γ_{max}, which is kept constant for all cases reported in this work.

If a state space model based on modal variables is to be used, the open-loop frequencies and mode shapes will be required. These can be calculated in a post processor to the finite element analysis code for given mass and stiffness matrices. In our program, the eigenvalues are first estimated using the Sturm sequence property of the eigenvalues [Bat], by successive bisection of the range containing the eigenvalues of interest. The eigenvectors of interest are then found using a forward iteration technique after applying a shift of the estimated eigenvalues as found above. Starting with an initial trial eigenvector populated by ones, the iteration technique should converge to the eigenvector corresponding to the eigenvalue for which the shift approximates. Care must be exercised in cases where "identical" eigenvalues are detected. In these cases, Gramm-Schmidt orthogonalization is used at the beginning and end of the iteration process to deflate the initial and final values of the eigenvectors of other eigenvectors associated with the repeated eigenvalue. Finally, after all eigenvectors of interest are extracted, the eigenvalue estimates are updated using the Rayleigh quotient of the appropriate eigenvectors. As a consequence, the eigenvalues will be accurate to second order in the error present in the eigenvectors [Nob].

The forward iteration method of finding eigenvectors finds one eigenpair at a time, and consequently numerical errors creep in and are transmitted and amplified as more and more eigenpairs are extracted. Since here we will either desire the entire eigenstructure for smaller systems, or a set number of the lowest frequency eigenpairs from which to construct a low-order design model for large systems, a subspace iteration technique to extract all the desired eigenpairs at once would probably be more appropriate. Errors in the eigenvectors can be very detrimental to the accuracy of the derivatives. The integrity of the derivatives is very important in any gradient based solution approach.

It is well known that the evaluation of eigenvectors is numerically difficult and subject to numerical error, especially for large systems and for systems with repeated eigenvalues. As was shown previously, the eigenvalues and eigenvectors, and their respective derivatives, are required for the evaluation of the derivatives of the state matrices with respect to the design variables in cases where a modal state space representation is used. Appendix B presents two methods for evaluating these derivatives, for systems both with and without repeated eigenvalues.

There is first a problem numerically in deciding when repeated eigenvalues are present. The number of repeated eigenvalues detected depends upon a specified detection cutoff value. Two eigenvalues are deemed identical if their relative difference is less than some value ε, which is specified by the user. The choice of ε is a non-trivial task since numerically two eigenvalues are never identical, and since we can generally not tell apriori the number and multiplicities of repeated eigenvalues present in a particular configuration. For example, for the DRAPER I structure with all structural design variables the same (equal area elements), and with ε set to 5×10^{-4}, four pairs of repeated eigenvalues are detected. However, if ε is set to 1×10^{-4}, only three pairs of repeated eigenvalues are detected.

If a repeated eigenvalue is not detected where in reality there is one, by setting ε too small, then the eigenvector obtained will be treated as unique when in fact it is not. Derivatives of eigenvectors for repeated eigenvalues only *exist* for specfic eigenvectors from the associated subspace. Therefore, failure to detect a repeated eigenvalue can result in large discontinuities in the associated eigenvector derivatives. Additionally, the eigenvector derivatives for *nonrepeated* eigenvalues are inversely proportional to the difference between that eigenvalue and all others (see Method 2 given in Appendix B). Therefore, a non-detected repeated eigenvalue will dominate the calculated derivative, and any errors in it will be greatly amplified. On the other hand, if a repeated eigenvalue is detected where in reality there is none, due to setting ε too large, an extra eigenvector will be included into the basis for the subspace of eigenvectors associated with that particular eigenvalue. This would increase the dimension of the associated subspace by adding a mutually orthogonal eigenvector. The associated *unique* eigenvectors from this subspace that have continuous derivatives would then include the effect of an eigenvector not associated with the repeated eigenvalue.

The calculation of eigenvectors and eigenvector derivatives for systems with repeated roots is a particularly troublesome and difficult task. Even though the majority of the time the system will probably not have repeated eigenvalues through symmetry, this can occur for example with symmetric structures subject to symmetric disturbances. In determining the derivatives of eigenvectors associated with repeated eigenvalues, it is assumed that the eigenvectors associated with a particular eigenvalue form a subspace, and that any linear combination of eigenvectors is also an eigenvector associated with that eigenvalue. Numerically however. this is not exactly true, due to the errors involved in finding these eigenvectors. As a consequence, even though the expressions given in Appendix B for derivatives of eigenvectors associated with repeated eigenvalues are analytically correct, our experience has shown that these derivatives are very sensitive to errors in the evaluated eigenvectors. These errors are carried into the gradients of the controller constraints, and hence into the approximate problem generation. When at an active controller constraint, solution of the approximate problem should move the design approximately along the constraint surfaces. However, at a design for which repeated eigenvalues are detected this may not occur because the errors in the gradients compromise the integrity of the approximate problem at this point.

One possible way to avoid these problems is to use a state space formulation based on physical variables. Here, only the mass and stiffness matrices are required, and these are calculated during the finite element analysis anyway. Such a solution to this problem is not practical for "large" systems however, where a model based on a subset of the modes will usually be used to reduce the computational and memory storage needs.

4.3 Results

Runs were made using CSOPT on the DRAPER I structure using initially an inverse design variable approximation for all constraint functions. The initial structure was defined with all structural design variables set at 10 in^2, and with the Lagrangian multipliers λ_u and λ_y set at 1.0. This set of initial conditions, summarized in Table 4.3, will be termed the symmetric set of initial conditions, for they specify a structure with a number of vibrational modes of the same frequency (repeated eigenvalues). A range of allowable expected output response (α^2) of 1×10^{-5} in^2 to 1×10^{-4} in^2 in steps of 1×10^{-4} in^2, and allowable expected control effort (β^2) of 50 lb^2 to S0 lb^2 in steps of 10 lb^2 were used.

Tables 4.4 and 4.6 summarize the resulting minimum weight in pounds found by CSOPT for each of these cases. The values in Table 4.4 correspond to the case where a state-space realization based on the modal displacements and velocities was used, hereafter called CASE A. The values in Table 4.6 correspond to the case where a state-space realization based on the physical nodal displacements and velocities was used, hereafter called CASE B. The specific value of γ_{min} used in each of the cases listed in Tables 4.4 and 4.6 are given in Tables 4.5 and 4.7 respectively.

We would intuitively expect two trends in the data displayed in Tables 4.4 and 4.6, as well as expecting the two tables to be the same. The optimum weight should decrease as the allowable control effort β^2 is increased at constant allowable output response α^2 (left to right across the table), and the optimum weight should decrease as the allowable output response α^2 is increased at constant allowable control effort β^2 (down the table). With reference to Table 4.4, we can see that this trend is observed in a macroscopic sense only, there being several examples where this trend is not observed. For example, considering the first column of Table 4.4, which corresponds to $\beta^2 = 50$ lb^2 for varying α^2, we see only two exceptions to the expected trends, these being at α^2 values of 6×10^{-5} and 9×10^{-5}. Similar results are observed in all other columns and rows of Table 4.4.

The results using the physical variable realization are more consistent than using the modal variables, although still not totally uniform. In Table 4.6, we find that there are only two locations where our expected trends do not seem to hold. In the case where $\beta^2 = 50$, the optimum weight for $\alpha^2 = 8 \times 10^{-5}$ of 1115.2 lbs is actually lower than might otherwise be expected considering the other values in this column. The only other exception is for $\beta^2 = 80$ and $\alpha^2 = 7 \times 10^{-5}$. One might then draw the conclusion that using the physical nodal variables as a basis for our state-space model results in a more consistent set of solutions, with respect to the optimum weight found, than does the case where the modal variables are used. We cannot however expand this conclusion to say that one state-space model produces better results (lower optimum weights) than the other. In fact, on this count the two results are very inconsistent. For example, for $\beta^2 = 50$, the optim' m weights found in CASE A are consistently lower than those found in CASE B, whereas for $\beta^2 = 80$, it is the CASE B results that are consistently better (or at least comparable). It might be pointed out that the results in CASE B were consistently easier to obtain, there being no need to alter the nominal value of γ_{min} (see Table 4.7 c.f. Table 4.5), and the number of global iterations required for convergence being consistently lower.

Some understanding of these contradictory results can be found by considering Tables 4.8 and 4.9, which give, for the two cases, the optimal element areas found for $\beta^2 = 50$ and for varying α^2. Also given in these tables are the number of global iterations required for convergence, the final values of the Lagrange multipliers (which then defines the LQR controller), and the initial value of the structural design variables (all the same for the symmetric set of initial conditions) at which the initial scaled system satisfies the constraints. Immediately apparent from Table 4.8 is a number of seemingly separate regions of the design space into which this structure has converged. For example, the final designs for $\alpha^2 = 5 \times 10^{-5}$ and $\alpha^2 = 7 \times 10^{-5}$ seem to be similar in relative structure. Here, "similar" refers to the relative sizing of the structural members, in that design variables that are "larger" in one design are "larger" in the other. Both these designs are however distinctly different from those for $\alpha^2 = 1 \times 10^{-5}$ and $\alpha^2 = 3 \times 10^{-5}$, which themselves are similar.

Considering Table 4.9 corresponding to a physical state space representation, we see a totally d'fferent phenomenom. Here, the solution algorithm seems to converge to the same region of the design space, with the exception of one case ($\alpha^2 = 8 \times 10^{-5}$). This corresponds to one of the cases for which the optimum weight does not fit into the pattern suggested by the other cases for the same allowable control effort, as mentioned above. This particular solution obviously lies in a different region of the design space than do the other solutions corresponding to other α^2 values. Also note that the number of global iterations required for convergence is very low in all cases except the one for which another region of the design space is encounted.

The conclusion seems to be that we are converging into different regions of the design space with our solution algorithm, and that there are numerous local minima. Several columns of Table 4.8 seem to define their own region of the design space, being dissimilar to any other column. In other words, our design space seems to have multidimensional corrugations leading to multiple local minima. The solutions will lie somewhere on the intersection hyperplane between the surface of constant allowable output response and the surface of constant allowable control effort.

This corrugated nature of the design space can be illustrated by considering the solutions obtained, for the same constraint case, when starting from different initial conditions. For the case of $\beta^2 = 75$ and $\alpha^2 = 1 \times 10^{-5}$, Tables 4.10 and 4.11 summarize the results of runs made when modal state space realizations and physical state space realizations are used respectively, when only the initial conditions are varied. The different initial conditions are defined by setting all structural elements equal except the first (element 1), to which is added a percentage of the size of other elements. Even with this limited variation in the initial conditions, there are seemingly many distinct regions in the design space into which the system may converge. A picture of the constraint surfaces as a one-dimensional slice of the multidimensional space will emerge if these optimal structures are varied into each other in a linear fashion, and the constraint values are calculated between each case. That is, the structural design variables and Lagrange multipliers are changed linearly from the optimal values in one case to those in another case. Then the constraint surfaces obtained would be those seen when travelling in a straight line between each successive point.

The results of such an analysis are shown in Figures 4.3 and 4.4 for the cases corresponding to those given in Tables 4.10 and 4.11 respectively. As expected, the weight varies linearly between the cases, but it is the constraint curves that are much more revealing. For example, considering Figure 4.3, one can see that between case 1 and case 2, there is a "ridge" of output response larger than the maximum allowable value. Similarly, the control effort first decreases, then also increases to a ridge of high value. This corresponds to a hump in the constraint surfaces between the two points in the design space. Assuming that we would see such behaviour when moving in every direction away from case 1 and case 2, rather than just in a direction between the two as shown in Figure 4.3, then the design points corresponding to these cases would represent local minima. In this situation, the design can become "trapped" in the such a locally convex region, causing the solution algorithm to converge to different points.

With reference to the same Figure 4.3, one can see that both the output response and control efforts are virtually constant between cases 2 and 3, while the weight increases slightly from 2053.0 lb to 2090.6 lb. This indicates that case 2 and case 3 actually represent the same optimal solution, with the difference being accounted for in the variance allowed by the convergence criteria used. The direction in the design space represented by the movement from case 2 to case 3 would lie in the intersection hyperplane of the surfaces of constant control effort and output response constraints, and would be at a shallow angle to the linear surface of constant weight.

Similar observations can be made considering Figure 4.4, where none of the cases appear to be representing the same optimal solution, although the solutions are very similar. Cases 1 and 2 in Figure 4.4 are only separated by a low ridge of constraint values, whereas cases 4 and 5 are separated by a high ridge. Although it is difficult to visualize, and impossible to sketch, the actual (in this case) 13-dimensional constraint surfaces and their 12-dimensional surface of intersection, travelling between specific points in the design space reduces the surfaces to manageable quantities. Figures 4.3 and 4.4 graphically illustrate a design space that is a very complicated function of the design variables, in which multiple local minima abound.

There are some other tests we can make on the hypothesis that we are becoming trapped in local minima. If the design is actually trapped in a local minimum, the solution should stay in the vicinity of that minimum if the problem is changed only slightly. That is, if a converged solution is used as the initial conditions for an optimization run where the constraint objectives are changed by a "small" amount, then the new problem should converge to a point that is "close to" the initial point.

Table 4.12 represents such a situation. Here, the solution was first obtained for the case where $\beta^2 = 50$ and $\alpha^2 = 1 \times 10^{-5}$, and where a modal state space realization and inverse design variable approximations were used. This converged solution was then used as the initial conditions for the cases $\beta^2 = 50$ and $\alpha^2 = 2 \times 10^{-5}$, and $\beta^2 = 60$ and $\alpha^2 = 1 \times 10^{-5}$. Moving down each column, and across the top row, of Table 4.12, the converged solution from the previous case was used as the initial condition for the new problem.

As can be seen from Table 4.12, the two expected trends in the data, as mentioned previously, are now observed without exception. The optimal solutions for the first two columns of Table 4.12, corresponding to cases where $\beta^2 = 50$ and $\beta^2 = 60$, are given in Tables 4.13 and 4.14 respectively. *All* the solutions now appear to be in the same local region of the design space, as evidenced by the relative sizing of the optimal structures. For example, note that in all converged designs, structural elements 9, 10, and 12 are at their lower gage limit of 0.1 in^2, and that the first structural element is the largest by far. These results test the hypothesis that designs are converging to local minima, and indicate that the local optima are real.

Convergence to local minima occurs regardless of whether physical or modal state space representations are used to model the structure. The final converged solution when a physical state space realization is used should also be optimal for the same constraint objective case if a modal state space realization is used instead. Such a situation is documented in Table 4.15. The first column in Table 4.15 gives the final converged solution for the case when $\beta^2 = 50$ and $\alpha^2 = 2 \times 10^{-5}$, and when a physical state space realization and inverse design variable approximations were used (c.f. column 2 of Table 4.8). When this solution was used as the initial condition for an optimization run for the same case, but when a modal state space realization was used rather than the physical realization, the results in the second column of Table 4.15 were obtained.

Note that the initial values for the control effort and output response constraints for the case when a modal state space realization was used (column 2 of Table 4.15) were not exactly 50 and 2×10^{-5} respectively as expected. Although the differences were small, they were numerically significant, and underscore the differences between physical and modal state space representations of the same structure. The differences come about due to the numerical errors involved in eigenvalue and eigenvector calculations. Because of this small difference, the structure is scaled up by a small amount, and the optimal solution is given by the variable values in the second column of Table 4.15. If *this* structure was then the initial condition for an optimization run where a physical state space representation was again used, the results obtained are given in the third column of Table 4.15. Once again, there is a small discrepancy between the initial constraint values and those expected, for the reason given above. Quite obviously however, all three designs given in Table 4.15 are the same local optimum point despite the small differences. Once again, this indicates that the apparent local optima do actually exist, and are not figments of a numerical imagination.

The only difference between the results from CASE A and CASE B runs is that modal and physically based state-space models respectively are used. The structure is initially symmetric, resulting in repeated eigenvalues. The final structures of CASE B listed in Table 4.9 are also symmetric with repeated eigenvalues (except that for $\alpha^2 = 8 \times 10^{-5}$). None of the optimal structures of CASE A listed in Table 4.8 are symmetric. In all our studies to date it seems that the solution algorithm in terms of physical state variables seems to be more predisposed to retain the symmetry than in the cases when a modal state space is used.

To illustrate "hat is happening, consider the iteration histories for the same run from CASE A and CASE B, in particular the case when $\beta^2 = 50$ and $\alpha^2 = 2 \times 10^{-5}$. The iteration histories are contained in Tables 4.16 and 4.17 for CASE A and CASE B respectively. From Table 4.16, we can see that during the first global iteration, when there are four detected repeated eigenvalues, and unlimited movement in the design space is allowed, the inverse design variable approximations used here are very poor approximations to the actual constraint behaviour. However, during the second global iteration, when we also allow virtually unlimited movement in the design space but now have no repeated eigenvalues, the approximations are much better.

These results are typical of all runs in CASE A. The combination of using an inverse design variable approximation and a modal state space representation seems to result in very poor approximate functions when repeated eigenvalues are present. We believe the reason for this is due to the errors present in the derivatives of eigenvectors associated with repeated eigenvalues, the causes of which were outlined in the previous section. However, even when there are no repeated eigenvalues present, we can have poor approximations due to cumulative errors in the gradients combined with an inverse design variable approximation formulation.

Using a physical nodal state space representation does not require calculation of eigenvalue and eigenvector derivatives, only the derivatives of the mass, stiffness and damping matrices. These are very easy to obtain given the element informations and the global and local elemental coordinate system definitions. The solution algorithm here seems to work very well and converge quickly, even in the case when repeated eigenvalues are present, as shown in Table 4.17. Again, the results indicated in Table 4.17 are typical for all runs in CASE B.

All of the results presented so far are for cases where inverse design variable approximations are used for the constraint functions. However, if hybrid design variable approximations are used, similar results and trends to those already noted are observed. The optimum weight found by CSOPT when hybrid design variable approximations and both modal and physical state space representations are used are given in Tables 4.18 and 4.20 respectively (hereafter termed CASE C and CASE D respectively). Tables 4.22 and 4.23 give the optimum designs for $\beta^2 = 50$ and varying α^2 for CASE C and CASE D respectively.

From Table 4.22 (CASE C), it is apparent that using hybrid design variable approximations and modal state space representations has led to optimal designs which are much more similar than was the case when inverse design variable approximations were used (compare to CASE A, Table 4.8). That is, the optimal designs listed in Table 4.22 for the most part seem to lie in the same region of the design space, since the relative sizes of the elements are similar. Of course, there are still exceptions, for example when $\alpha^2 = 7 \times 10^{-5}$ and $\alpha^2 = 9 \times 10^{-5}$. These two designs are the two that might seem anomolous when considering the $\beta^2 = 50$ column of Table 4.18. Final converged weights however, seem to be much larger when using hybrid approximations than were obtained using inverse approximations. This can be explained by the fact that the hybrid approximations give a better approximation of the constraint surfaces than do the inverse approximations. Therefore, less scaling to the constraint surfaces need be done, and the design tends to converge quickly to a local minima close to the specified initial conditions. Even shallow local minima will trap the design easily because the approximations are predicting the actual constraint behaviour very well. Of course, if the initial point is in an advantageous point in the design space, lower weight solutions may be obtained using hybrid approximations than were obtained using inverse approximations, as seen in some cases in Table 4.23.

If a physical state space representation is used, then using either the inverse or hybrid design variable approximations seem for the most part to give optimal designs for varying α^2 and β^2 that lie in the same region of the design space. In fact, comparing Tables 4.6 and 4.20, many of the optimum weights are basically identical regardless of whether inverse or hybrid design variable approximations are used, even though the actual optimum design variables may not be the same (compare Tables 4.9 and 4.23). Once again however, referring to Table 4.23 (CASE D), there are exceptions, notably for $\alpha^2 = 2 \times 10^{-5}$ and $\alpha^2 = 6 \times 10^{-5}$.

Our conclusion is that this procedure has achieved only limited success. Solutions can be obtained, but no guarantee can be given that these solutions will be "good", and no algorithmic way exists to decide how to go about getting a better (lower weight) solution. There does seem to be some advantages to be gained in *not* attempting to be as accurate as possible in approximating the constraint functions. The corrugated nature of the design space makes the problem very difficult, and these difficulties can only get worse as the dimensionality increases.

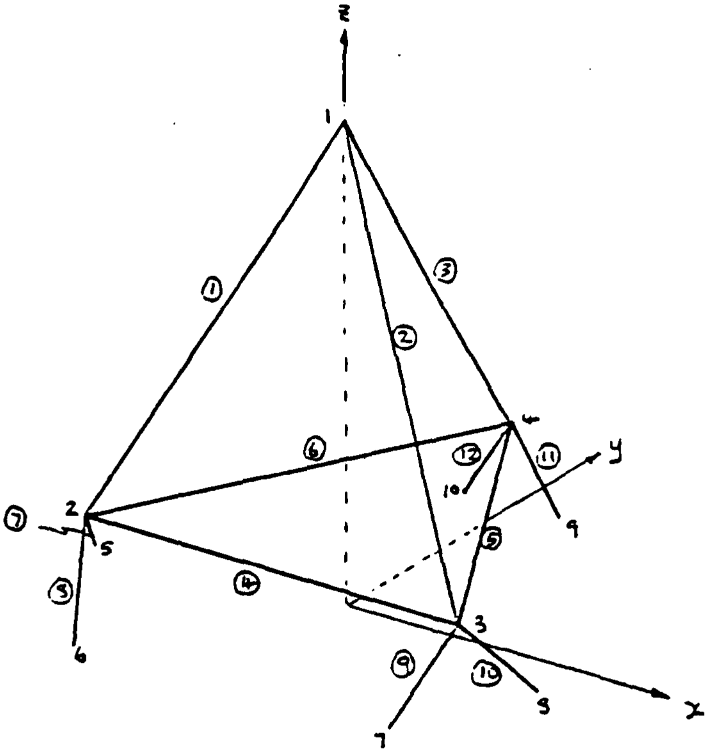

Figure 4.1: The DRAPER I Structure

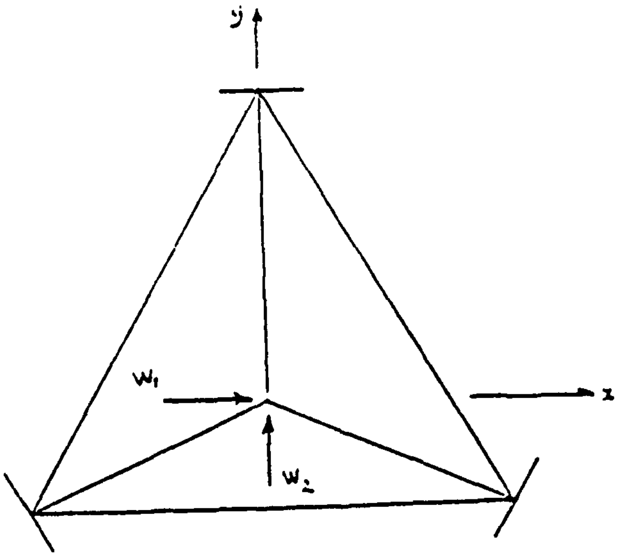

Figure 4.2: The disturbance model

Node	x (ft)	y (ft)	z (ft)
1	0.0	0.0	10.165
2	-5.0	-2.887	2.0
3	5.0	-2.887	2.0
4	0.0	5.7735	2.0
5	-6.0	-1.1547	0.0
6	-4.0	-4.6188	0.0
7	4.0	-4.6188	0.0
8	6.0	-1.1547	0.0
9	2.0	5.7735	0.0
10	-2.0	5.7735	0.0

Table 4.1: Nodal Positions for the DRAPER I Structure

Element	Node 1	Node 2
1	1	2
2	1	3
3	1	4
4	2	3
5	3	4
6	2	4
7	2	5
8	2	6
9	3	7
10	3	8
11	4	9
12	4	10

Table 4.2: Truss Element Connectivity Data for DRAPER I Structure

Element	Initial value (in^2)
1	10.0
2	10.0
3	10.0
4	10.0
5	10.0
6	10.0
7	10.0
8	10.0
9	10.0
10	10.0
11	10.0
12	10.0

$$\lambda_u = 1.0$$

$$\lambda_y = 1.0$$

Table 4.3: Symmetric Set of Initial Conditions

α^2 ($\times 10^{-5}$)	β^2			
	50	60	70	80
1	2347.1	2991.4	2107.4	2924.9
2	2077.3	2030.8	1914.4	1289.7
3	1878.0	1658.9	1472.9	1216.8
4	1548.3	1409.7	1653.2	1021.5
5	1418.9	1319.2	1459.5	937.4
6	1460.6	1209.5	964.9	1009.8
7	1209.0	1076.0	928.7	1052.7
8	1066.7	1214.1	987.9	790.2
9	1126.3	940.5	854.0	720.1
10	971.9	1035.1	766.4	970.6

Table 4.4: Optimal weight using a modal state-space realization, the symmetric set of initial conditions, and inverse design variable approximations. (CASE A)

α^2 ($\times 10^{-5}$)	β^2			
	50	60	70	80
1	1.25	1.25	1.25	1.25
2	1.20	1.25	1.25	1.25
3	1.20	1.25	1.25	1.20
4	1.20	1.25	1.25	1.15
5	1.20	1.25	1.25	1.15
6	1.20	1.25	1.15	1.15
7	1.20	1.25	1.15	1.15
8	1.20	1.25	1.15	1.15
9	1.10	1.20	1.15	1.15
10	1.10	1.20	1.15	1.30

Table 4.5: γ_{min} used to obtain the values in Table 4.4 (CASE A)

α^2 ($\times 10^{-5}$)	β^2			
	50	60	70	80
1	3241.4	2717.8	2570.9	1821.8
2	2395.4	1921.0	1626.5	1362.4
3	2012.1	1623.3	1357.3	1141.4
4	1779.9	1429.2	1241.7	999.7
5	1618.1	1299.0	1088.5	932.5
6	1496.8	1202.2	1009.8	908.1
7	1400.9	1128.6	948.9	955.2
8	1115.2	1065.7	893.9	787.1
9	1260.1	1014.5	851.7	724.2
10	1206.2	970.4	814.4	673.5

Table 4.6: Optimal weight using a physical state-space realization, the symmetric set of initial conditions, and inverse design variable approximations. (CASE B)

α^2 ($\times 10^{-5}$)	β^2			
	50	60	70	80
1	1.25	1.25	1.25	1.25
2	1.25	1.25	1.25	1.25
3	1.25	1.25	1.25	1.25
4	1.25	1.25	1.25	1.25
5	1.25	1.25	1.25	1.25
6	1.25	1.25	1.25	1.25
7	1.25	1.25	1.25	1.25
8	1.25	1.25	1.25	1.25
9	1.25	1.25	1.25	1.25
10	1.25	1.25	1.25	1.25

Table 4.7: γ_{min} used to obtain the values in Table 4.6 (CASE B)

	α^2 ($\times 10^{-5}$)									
	1	2	3	4	5	6	7	8	9	10
Scaled DV	90.437	63.947	52.212	45.213	40.441	36.918	34.180	31.973	29.979	28.594
Final DV										
1	125.893	73.091	70.557	49.717	43.868	25.936	32.194	9.723	16.757	20.331
2	23.336	53.545	36.664	21.137	23.151	30.650	21.255	27.425	32.495	38.508
3	45.632	15.268	14.833	35.217	21.510	28.247	20.736	28.618	16.565	11.434
4	6.389	10.704	5.599	6.318	4.957	11.168	5.141	4.083	8.798	2.041
5	10.973	5.375	12.207	4.213	9.637	10.922	7.713	6.737	4.348	2.369
6	9.800	6.209	6.953	5.365	4.914	6.795	4.260	3.815	6.692	2.934
7	17.627	8.610	10.775	5.638	9.310	0.101	8.299	0.100	0.100	0.152
8	17.719	0.124	8.802	0.150	9.901	2.501	9.363	0.100	9.628	0.885
9	0.100	13.250	0.100	0.100	0.100	0.245	0.100	6.482	1.332	3.242
10	0.100	9.332	0.711	0.313	8.307	9.107	8.183	8.323	0.100	5.016
11	18.206	0.100	13.746	9.473	8.352	7.744	7.379	8.218	7.737	2.532
12	0.100	0.100	0.100	9.266	0.100	8.583	0.100	6.782	10.099	0.100
λ_u	1.6710	1.4574	1.2620	1.4851	0.8632	0.7522	0.8544	0.7937	0.9117	1.7622
λ_y	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
iter. for convergence	44	19	15	32	32	21	18	22	22	49

Table 4.8: Optimal design variables for $\beta^2 = 50$ in CASE A

	α^2 ($\times 10^{-5}$)									
	1	2	3	4	5	6	7	8	9	10
Scaled DV	108.789	76.924	62.807	54.392	48.650	44.411	41.117	38.461	36.261	34.401
Final DV										
1	84.594	61.454	51.563	45.089	40.826	37.634	35.206	35.938	31.444	30.105
2	84.594	61.454	51.563	45.089	40.826	37.634	35.206	36.090	31.444	30.105
3	84.306	61.524	51.036	45.095	40.707	37.519	35.082	7.280	31.387	30.066
4	5.546	5.008	4.317	4.207	4.121	3.425	3.669	2.885	3.580	3.369
5	5.454	5.001	4.512	4.338	4.095	4.177	3.703	2.866	3.494	3.349
6	5.454	5.001	4.512	4.338	4.095	4.177	3.703	3.456	3.494	3.349
7	0.100	0.100	0.100	0.100	0.100	0.100	0.100	7.681	0.100	0.100
8	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.359	0.100	0.100
9	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.384	0.100	0.100
10	0.100	0.100	0.100	0.100	0.100	0.100	0.100	7.000	0.100	0.100
11	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100
12	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100
λ_u	2.8377	2.8718	2.8898	2.9014	2.9065	2.9081	2.9088	1.8801	2.9002	2.8976
λ_y	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
iter. for convergence	5	8	7	6	6	5	6	23	8	7

Table 4.9: Optimal design variables for $\beta^2 = 50$ in CASE B

	Case 1	Case 2	Case 3	Case 4	Case 5
Final Wt.	2451.3	2053.0	2090.6	1915.6	2067.0
Final DV					
1	73.849	41.076	55.595	38.306	60.319
2	70.699	49.211	55.391	68.734	48.760
3	13.262	74.820	58.403	44.107	50.104
4	16.100	2.146	1.6065	1.227	2.841
5	7.730	1.843	1.4806	1.612	2.169
6	8.068	1.819	1.5676	3.161	2.834
7	3.864	0.100	0.100	0.100	0.170
8	20.163	0.100	0.100	0.451	0.100
9	21.568	0.100	0.100	7.939	8.984
10	5.694	0.100	0.100	0.100	0.100
11	0.100	0.100	0.100	0.100	0.100
12	0.100	0.100	0.100	0.100	9.014
λ_u	0.9765	3.1055	3.3391	2.6755	2.5236

For all $j \neq 1$, the initial conditions are:
Case 1: $p_1 = p_j$
Case 2: $p_1 = p_j + 3\%$
Case 3: $p_1 = p_j + 6\%$
Case 4: $p_1 = p_j + 10\%$
Case 5: $p_1 = p_j + 50\%$

For all cases, $(\lambda_u)_0 = 1.0$, $(\lambda_y)_0 = 1.0$

Table 4.10: Optimal values when $\beta^2 = 75$ and $\alpha^2 = 1 \times 10^{-5}$ for differing initial conditions, when using a modal state space model

	Case 1	Case 2	Case 3	Case 4	Case 5
Final Wt.	2006.6	1952.6	1940.6	1943.7	1875.5
Final DV					
1	52.392	47.085	28.699	32.257	63.124
2	52.392	48.486	64.095	64.893	39.146
3	53.498	59.061	55.951	46.706	41.904
4	2.894	3.565	3.370	5.534	3.865
5	2.933	2.486	4.261	4.591	0.901
6	2.933	1.621	3.601	4.992	3.438
7	0.100	0.556	0.100	0.237	0.102
8	0.100	0.100	0.407	0.100	0.102
9	0.100	0.100	5.331	0.100	0.100
10	0.100	0.100	0.100	0.100	6.512
11	0.100	0.100	0.100	0.100	6.910
12	0.100	0.506	0.100	9.956	0.100
λ_u	2.8362	2.8044	2.3156	2.1606	2.3039

For all $j \neq 1$, the initial conditions are:
Case 1: $p_1 = p_j$
Case 2: $p_1 = p_j + 3\%$
Case 3: $p_1 = p_j + 6\%$
Case 4: $p_1 = p_j + 10\%$
Case 5: $p_1 = p_j + 50\%$

For all cases, $(\lambda_u)_0 = 1.0$, $(\lambda_y)_0 = 1.0$

Table 4.11: Optimal values when $\beta^2 = 75$ and $\alpha^2 = 1 \times 10^{-5}$ for differing initial conditions, when using a physical state space model

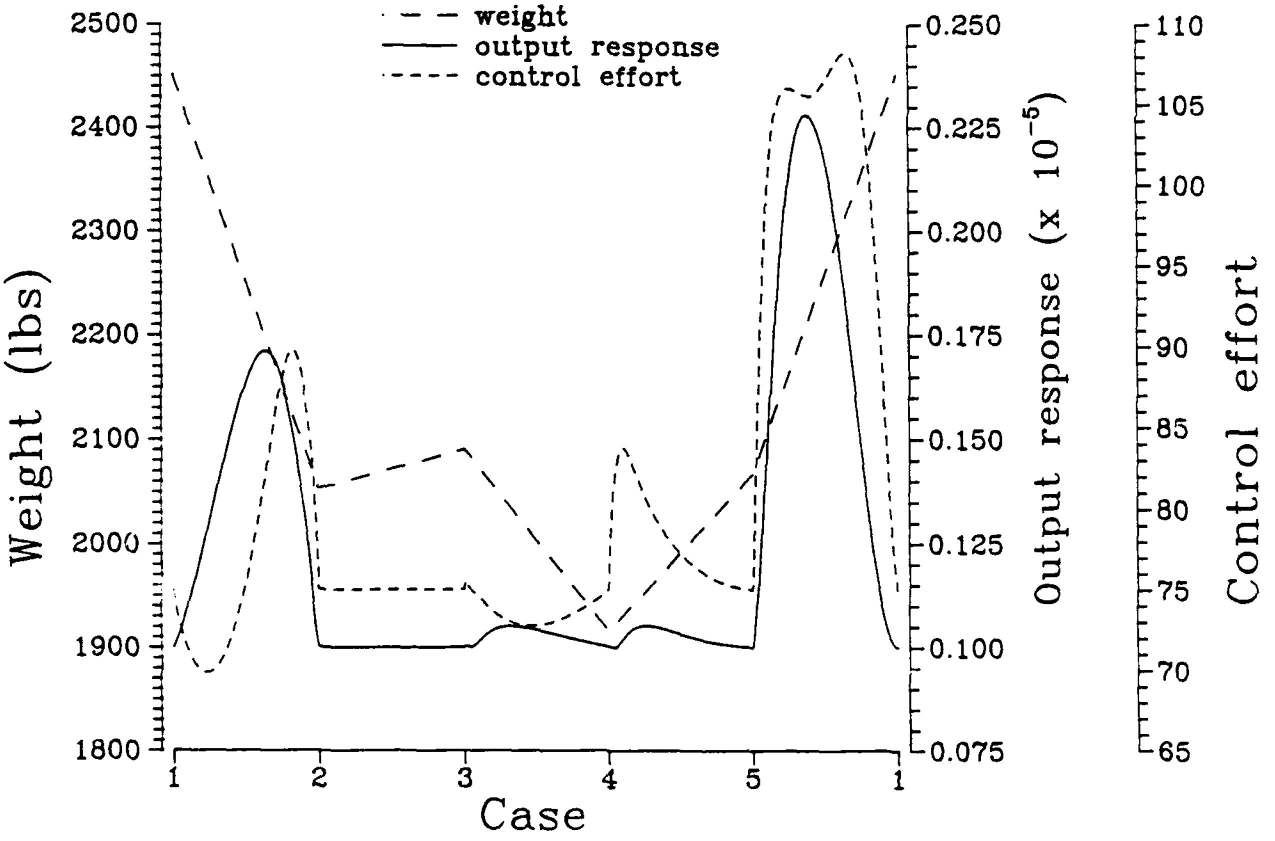

Figure 4.3: Constraint surfaces between cases listed in Table 4.10

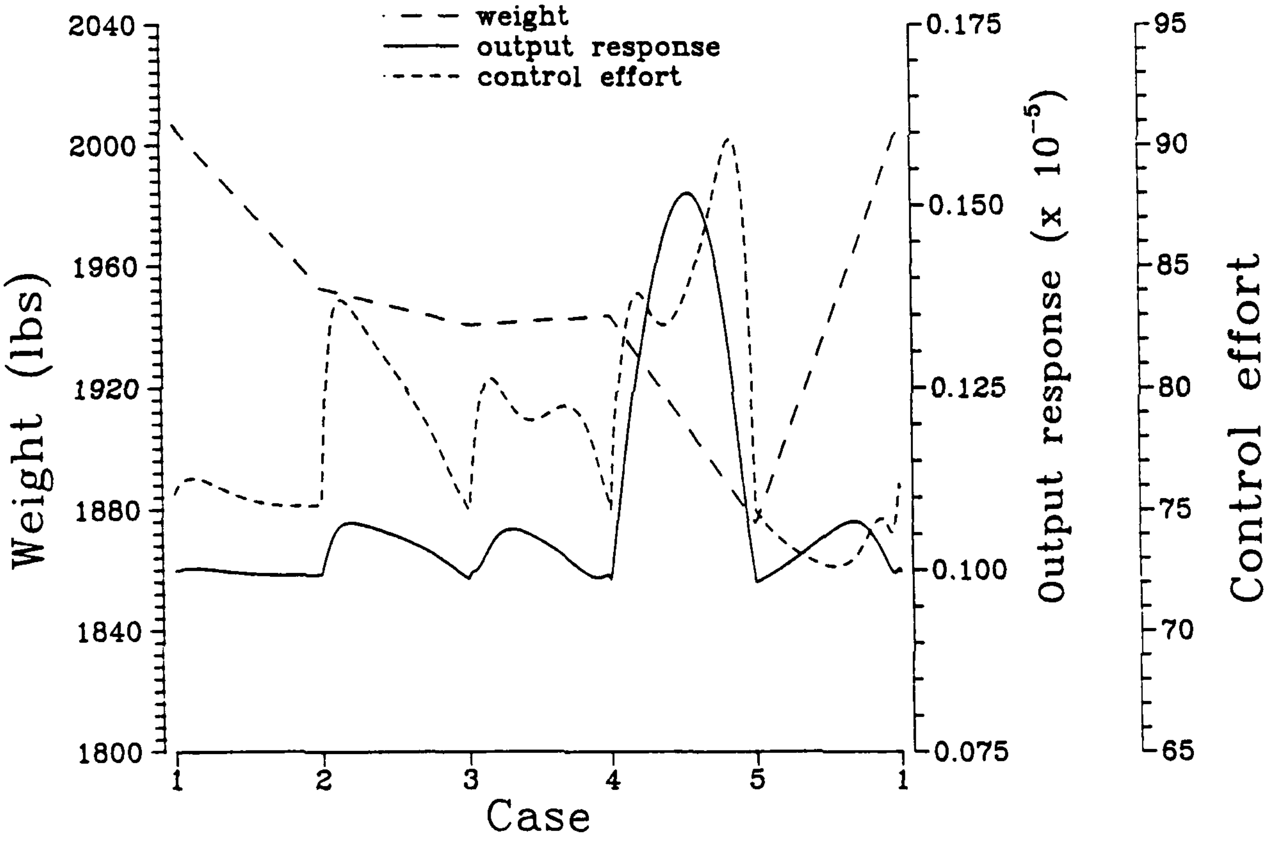

Figure 4.4: Constraint surfaces between cases listed in Table 4.11

α^2 ($\times 10^{-5}$)	β^2			
	50	60	70	80
1	2847.1	2466.0	2166.6	1969.2
2	2045.9	1782.1	1579.7	1443.2
3	1685.6	1478.8	1327.5	1197.2
4	1470.9	1298.0	1162.0	1052.0
5	1325.2	1174.2	1049.3	952.1
6	1217.0	1083.2	965.9	880.0
7	1134.1	1012.4	901.4	823.8
8	1065.7	955.3	849.4	776.5
9	1014.5	907.9	806.4	735.3
10	961.3	861.0	770.0	700.2

Table 4.12: Optimal weight using a modal state-space realization and inverse design variable approximations, where the initial condition for each case is the converged solution from the previous case.

	α^2 ($\times 10^{-5}$)				
	1	2	3	4	5
Final DV					
1	125.893	82.406	66.777	57.864	50.877
2	23.336	19.075	16.010	14.170	13.257
3	45.632	36.194	29.987	26.286	23.752
4	6.389	6.372	5.386	4.736	4.105
5	10.973	8.135	6.487	5.952	5.429
6	9.800	8.558	7.433	6.631	6.031
7	17.627	12.148	12.085	9.342	11.208
8	17.719	10.689	8.446	6.718	5.422
9	0.100	0.100	0.100	0.100	0.100
10	0.100	0.100	0.100	0.100	0.100
11	18.206	11.326	8.820	8.172	7.764
12	0.100	0.100	0.100	0.100	0.100
λ_u	1.6710	1.6281	1.6293	1.6325	1.6273
λ_y	1.0	1.0	1.0	1.0	1.0
iter. for convergence	44	16	2	2	2

	α^2 ($\times 10^{-5}$)				
	6	7	8	9	10
Final DV					
1	46.745	42.850	40.259	37.112	33.742
2	12.302	11.800	11.108	11.403	11.179
3	21.926	20.458	19.239	18.610	17.761
4	3.996	3.621	3.446	4.045	3.745
5	5.144	4.708	4.451	3.751	4.167
6	5.556	5.241	4.926	4.681	4.634
7	7.501	9.330	8.344	7.053	6.621
8	5.286	4.390	4.167	5.319	5.070
9	0.100	0.100	0.100	0.100	0.100
10	0.100	0.100	0.100	0.100	0.100
11	7.237	6.603	6.216	4.678	5.357
12	0.100	0.100	0.100	0.100	0.100
λ_u	1.6243	1.6163	1.6167	1.5670	1.5540
λ_y	1.0	1.0	1.0	1.0	1.0
iter. for convergence	2	2	2	4	3

Table 4.13: Optimal design variables for $\beta^2 = 50$ cases given in Table 4.12

	α^2 $(\times 10^{-5})$									
	1	2	3	4	5	6	7	8	9	10
Final DV										
1	114.695	82.224	68.219	59.872	54.153	49.952	46.684	44.047	41.861	39.029
2	15.953	11.936	9.905	8.695	7.866	7.258	6.784	6.402	6.086	6.565
3	39.232	28.262	23.448	20.579	18.614	17.170	16.047	15.141	14.390	13.696
4	5.278	4.230	3.510	3.082	2.789	2.575	2.408	2.273	2.162	2.445
5	9.266	6.601	5.478	4.809	4.351	4.015	3.754	3.543	3.369	3.122
6	9.229	6.831	5.670	4.978	4.505	4.157	3.887	3.668	3.488	3.280
7	14.382	10.519	8.747	7.686	6.966	6.433	6.018	5.683	5.406	4.564
8	12.023	8.285	6.865	6.016	5.421	4.982	4.637	4.355	4.114	3.225
9	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100
10	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100
11	15.158	10.676	8.848	7.760	7.010	6.461	6.034	5.691	5.406	4.685
12	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100	0.100
λ_u	1.5486	1.5754	1.5948	1.6090	1.6202	1.6300	1.6383	1.6453	1.6513	1.6330
λ_y	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
iter. for convergence	20	5	2	2	2	2	2	2	2	4

Table 4.14: Optimal design variables for $\beta^2 = 60$ cases given in Table 4.12

	state space realization		
	physical	modal	physical
Initial output response ($\times 10^{-5}$)	—	2.0575	1.943
Initial control effort	—	50.3162	49.876
Initial weight	—	2077.3	2100.9
Final DV			
1	73.091	72.075	70.662
2	53.545	53.401	52.344
3	15.268	15.355	15.203
4	10.704	11.849	11.718
5	5.375	4.579	4.856
6	6.209	6.921	7.317
7	8.610	10.486	10.259
8	0.124	0.152	0.125
9	13.250	16.378	15.847
10	9.332	11.299	9.687
11	0.100	0.100	0.100
12	0.100	0.100	0.100
Final weight	2077.3	2100.9	2067.8
λ_u	1.4574	1.4140	1.4510
iter. for convergence	19	3	2

Table 4.15: Optimal designs for $\beta^2 = 50$ and $\alpha^2 = 2 \times 10^{-5}$ when modal and physical state space realizations are used.

			after approx. prob. solution			scaling for	number of
iter.	initial	γ	weight	E[$\boldsymbol{u}^T R\boldsymbol{u}$]	E[$\boldsymbol{y}_d^T W\boldsymbol{y}_d$]	constraint	repeated
no.	weight				($\times10^{-5}$)	satisfaction	eigenvalues
—	923.6	—	—	450.6	27.52	6.395	4
1	5906.5	1000	4739.4	1313.3	27.25	12.39	0
2	58726.9	368	57118.7	49.33	1.944	0.9858	0
3	56306.8	136	34800.1	677.9	0.810	1.367	0
4	47556.1	50.9	44924.6	121.8	2.159	1.465	0
5	65834.9	19.5	64824.7	48.77	1.943	0.9758	0
6	63258.9	7.93	13340.9	30.77	0.951	0.5307	0
7	8333.5	3.68	3092.7	66.99	2.154	1.279	0
8	3954.5	2.11	3125.4	47.45	1.844	0.9235	0
9	2886.1	1.54	2586.5	50.16	2.009	1.005	0
10	2598.2	1.32	2442.3	49.64	1.973	0.9877	0
11	2412.2	1.25	2317.2	50.04	1.997	0.9999	0
12	2316.9	1.22	2255.3	49.50	1.965	0.9845	0
13	2220.5	1.21	2192.4	49.72	2.004	0.9971	0
14	2186.0	1.20	2153.1	49.64	1.992	0.9931	0
15	2138.0	1.20	2113.8	49.92	1.996	0.9979	0
16	2109.3	1.20	2098.7	49.72	1.996	0.9950	0
17	2088.2	1.20	2087.9	49.85	2.000	0.9970	0
18	2083.5	1.20	2084.7	49.82	1.999	0.9973	0
19	2079.1	1.20	2080.3	49.89	2.000	0.9986	0
—	2077.3	—	—	49.99	2.000	—	0

Table 4.16: Iteration history for $\beta^2 = 50$ and $\alpha^2 = 2 \times 10^{-5}$ for CASE A

			after approx. prob. solution			scaling for	number of
iter.	initial	γ	weight	$\mathrm{E}[\boldsymbol{u}^T R \boldsymbol{u}]$	$\mathrm{E}[\boldsymbol{y}_d^T W \boldsymbol{y}_d]$	constraint	repeated
no.	weight				$(\times 10^{-5})$	satisfaction	eigenvalues
—	923.6	—	—	533.7	28.10	7.692	4
1	7105.1	1000	6933.2	46.97	1.461	0.8402	4
2	5825.1	368	1644.6	97.13	0.9261	1.695	0
3	2788.9	136	2287.5	51.95	2.092	1.080	0
4	2469.9	51.0	2399.1	50.14	2.000	1.004	0
5	2409.2	19.5	2397.2	50.04	2.001	—	3
6	2397.2	7.98	2399.9	50.00	1.999	—	3
7	2399.7	3.72	2395.8	50.01	2.000	—	3
8	2395.8	2.16	2395.4	50.01	2.000	—	3
—	2395.4	—	—	50.01	2.000	—	3

Table 4.17: Iteration history for $\beta^2 = 50$ and $\alpha^2 = 2 \times 10^{-5}$ for CASE B

	β^2	
α^2 ($\times 10^{-5}$)	50	60
1	5086.7	5222.9
2	3545.8	3717.4
3	2943.0	2719.9
4	2606.4	2299.7
5	2342.9	2223.2
6	2188.7	1995.8
7	1372.0	1302.3
8	1838.8	1720.4
9	1094.5	1620.1
10	1649.4	1556.5

Table 4.18: Optimal weight using a modal state-space realization, the symmetric set of initial conditions, and hybrid design variable approximations. (CASE C)

	β^2	
α^2 ($\times 10^{-5}$)	50	60
1	1.25	1.25
2	1.25	1.25
3	1.25	1.25
4	1.25	1.25
5	1.25	1.25
6	1.25	1.25
7	1.25	1.25
8	1.25	1.25
9	1.25	1.25
10	1.25	1.25

Table 4.19: γ_{min} used to obtain the values in Table 4.18 (CASE C)

	β^2	
α^2 $(\times 10^{-5})$	50	60
1	4081.2	2598.2
2	2069.7	1951.7
3	2557.3	1619.8
4	2231.5	1428.2
5	1522.2	1272.9
6	1401.8	1206.5
7	1656.5	1127.0
8	1573.4	939.4
9	1461.8	978.2
10	1390.7	957.7

Table 4.20: Optimal weight using a physical state-space realization, the symmetric set of initial conditions, and hybrid design variable approximations. (CASE D)

	β^2	
α^2 $(\times 10^{-5})$	50	60
1	1.25	1.25
2	1.25	1.25
3	1.25	1.25
4	1.25	1.25
5	1.25	1.25
6	1.25	1.25
7	1.25	1.25
8	1.25	1.25
9	1.25	1.25
10	1.25	1.25

Table 4.21: γ_{min} used to obtain the values in Table 4.20 (CASE D)

	α^2 ($\times 10^{-5}$)				
	1	2	3	4	5
Scaled DV	90.437	63.947	52.212	45.213	40.441
Final DV					
1	116.537	81.263	67.488	59.206	52.214
2	87.551	63.283	51.825	45.141	40.642
3	61.273	42.350	36.864	31.630	28.695
4	58.520	39.845	35.986	30.204	27.994
5	32.627	23.613	19.781	15.285	13.356
6	9.427	6.661	4.829	4.572	4.043
7	28.100	17.570	14.569	15.321	14.587
8	57.705	40.663	22.056	25.805	23.107
9	46.652	32.261	23.649	24.185	21.667
10	33.016	20.119	17.652	21.179	18.471
11	39.318	25.292	22.653	23.688	22.125
12	0.100	0.100	0.100	0.100	0.101
λ_u	0.4020	0.4287	0.4319	0.3717	0.3671
λ_y	1.0	1.0	1.0	1.0	1.0
iter. for convergence	30	21	18	21	17

	α^2 ($\times 10^{-5}$)				
	6	7	8	9	10
Scaled DV	36.918	34.180	31.973	29.979	28.594
Final DV					
1	47.435	29.891	41.464	18.029	37.093
2	37.598	27.263	31.927	23.017	28.348
3	26.607	22.176	22.521	25.037	20.347
4	28.581	11.703	21.748	5.455	19.389
5	10.641	9.119	10.791	5.107	9.535
6	3.741	6.946	3.188	4.202	2.903
7	12.179	6.183	10.791	0.100	9.717
8	24.414	5.617	18.756	8.244	16.051
9	21.866	7.299	15.194	8.509	15.714
10	18.713	0.292	14.827	0.100	13.490
11	20.965	6.080	16.910	9.849	15.053
12	0.100	0.100	0.100	9.836	0.100
λ_u	0.3447	0.5782	0.3781	0.7831	0.3719
λ_y	1.0	1.0	1.0	1.0	1.0
iter. for convergence	12	33	20	21	24

Table 4.22: Optimal design variables for $\beta^2 = 50$ in CASE C

	α^2 $(\times 10^{-5})$									
	1	2	3	4	5	6	7	8	9	10
Scaled DV	108.789	76.924	62.807	54.392	48.650	44.411	41.117	38.461	36.261	34.401
Final DV										
1	112.468	25.786	56.915	49.314	48.663	45.880	37.259	34.749	32.774	31.123
2	112.468	28.096	56.915	49.314	48.663	45.880	37.259	34.749	32.774	31.123
3	112.459	101.308	56.911	49.310	15.832	12.703	37.256	34.745	32.770	31.120
4	0.842	4.676	13.774	12.225	3.667	4.152	8.534	8.733	7.631	7.326
5	0.842	3.696	13.784	12.393	4.634	3.771	8.575	8.736	7.649	7.344
6	0.842	3.951	13.784	12.393	4.634	3.766	8.575	8.736	7.649	7.349
7	0.100	0.100	0.608	0.591	1.125	0.970	0.347	0.400	0.319	0.313
8	0.100	0.100	0.604	0.591	0.100	0.100	0.346	0.383	0.308	0.303
9	0.100	0.100	0.604	0.591	0.100	0.100	0.346	0.383	0.308	0.303
10	0.100	0.100	0.608	0.591	1.123	0.970	0.347	0.400	0.319	0.313
11	0.100	8.171	0.598	0.591	0.100	0.100	0.342	0.393	0.284	0.283
12	0.100	8.960	0.598	0.591	0.100	0.100	0.342	0.393	0.284	0.283
λ_u	3.9224	2.3377	2.2676	2.1723	2.2227	2.2502	2.3706	2.2168	2.3701	2.3404
λ_y	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
iter. for convergence	13	32	4	4	15	16	4	6	5	5

Table 4.23: Optimal design variables for $\beta^2 = 50$ in CASE D

Chapter 5

Conclusions and Recommendations

The problem of the integrated control/structure design optimization of large flexible structures has been investigated for the case of full state feedback control of structures subject to a stochastic disturbance input model. Due to the special nature of the controller and problem formulation, the controller design variables (elements of K) were found to be given by the solution of an associated LQR problem. Therefore, these variables are not treated explicitly as design variables in the optimization procedure, but rather sized by a scaling procedure developed to identically satisfy the constraints. A research computer program was developed to solve the problem using a sequential approximations technique and commercially available nonlinear programming code. Very efficient expressions for the required gradients were derived that significantly reduced the computational burden of the solution algorithm.

The solution algorithm was applied to the DRAPER I tetrahedral truss strucutre for a range of constraint objectives. Convergence could be obtained, and an optimal solution found, in every case attempted in this work. However, the problem was found to have many local minima. Although this is generally the case for non-globally convex problems, it is the dominant factor in this particular problem. The solution method was very sensitive to the various parameters that must be rather arbitrarily chosen, such as move-limits and initial conditions. Altering either of these two, for example, can result in the design converging to distinctly different regions of the design space, with a resultant difference in the optimal weight found.

More consistent results were obtained, when varying the design constraints, for cases when a physically based state space realization was used than those cases where a modal state space realization was used, although often the optimal weight found was larger. Here, "consistent" refers to the similarity of optimal designs when the constraints were changed. This is due to the difficulties encountered in evaluating accurate derivatives for eigenvectors associated with repeated eigenvalues. The sensitivity of these gradients to inaccuracies in the eigenvectors in cases where repeated eigenvalues are detected means that when using a modally based state space system, the solution is biased away from symmetric designs. Cases when a physically based state space system was used indicated no such biasing, and in fact many optimum designs in these cases exhibited symmetry (repeated eigenvalues).

There was a great deal of variation in the optimal designs when modal state space realizations and inverse design variable approximations were used, due largely to the first few iteration steps. Initial steps tended to be quite large due to: (i) large move-limits, (ii) the inaccuracies noted in the repeated eigenvector derivatives, and (iii) the significant scaling of the structure due to inaccuracies in the inverse approximations. This caused the design to "jump" around in the design space for a few iterations, not unlike the process of "simulated annealing", and the optimum weight obtained was more often than not lower than cases where the initial iterations were "well-behaved", as in the case of a physical state space representation and/or hybrid design variable approximations.

For this reason, and because of the apparent existance of so many local minima, it is recommended that future work should investigate various options for resolving the best solutions. Systematic solutions using random initial conditions, or the intentional inclusion of a random perturbing factor as in the "simulated annealing" procedure, may be required to obtain solutions.

This work concentrated on the case when full state feedback control is used. However, in most designs of practical significance, full state feedback is probably not practical. Future work should include studies on different controller methodologies, and how they affect the solution algorithm. Examples of large dimension must be attempted and solved to give confidence that this solution process is viable in realistic design situations.

Even with these problems, our conclusion is that the combined control/structure optimization problem has a high payoff potential, and should continue to be explored. The ability of the procedure to trade-off structural mass verses controller energy is an important tool for coupling structural system and controller design synthesis, and will be crucial in the design of realistic structures for widespread application in the coming decades.

Bibliography

[Ada] Adamian, Armen, "Integrated Control/Strucutre Design and Robustness", Ph.D. Dissertation, UCLA, 1986.

[Bat] Bathe, Klaus-Jurgen and Wilson, Edward L., *Numerical Methods in Finite Element Analysis*, Prentice-Hall Inc., 1976.

[Bod] Bodden, D.S. and Junkins, J.L., "Eigenvalue Optimization Algorithms for Structure/Controller Design Iterations", *J. Guidance and Control*, Vol. 8, No. 6, Nov.–Dec. 1985, pp 697–706.

[Can] Canfield, R.A., Grandhi, R.V. and Venkayya, V.B., "Comparison of Optimization Algorithms for Large Structures", AFWAL-TM-86-204-FIBR, Wright-Patterson Air Force Base, 1986.

[Cra] Crane, R.L., "Solution of the General Nonlinear Programming Problem with Subroutine VMCON", ANL-80-64, Argonne National Laboratory, Argonne, IL.

[Fox] Fox, R.L. and Kapoor, M.P., "Rates of Change of Eigenvalues and Eigenvectors",*AIAA Journal*, Vol. 6, Dec. 1968, pp 2426–2429.

[Gra-1] Grandhi, R.V., Thareja, R. and Haftka, R.T., "NEWSUMT–A: A General Purpose Program for Constrained Optimization Using Constraint Approximations", *J. of Mechanisms, Transmissions and Automation in Design*, Vol. 107, March 1985, pp 94–99.

[Gra-2] Grandhi, R.V. and Venkayya, V.B., "Structural Optimization with Frequency Constraints", *AIAA Journal*, Vol 26, No. 7, July 1988, pp 858–866.

[Gus] Gustafson, C.L., Aswani, M., Doran, A.L. and Tseng, G.T.,"ISAAC (Integrated Structural Analysis and Control) via Continuum Modeling and Distributed Frequency Domain Techniques", Presented at the JPL Workshop on Identification and Control of Flexible Space Structures, San Diego, California, June 4–6, 1984.

[Haf-1] Haftka, Raphael T., "Design for Temperature and Thermal Buckling Constraints Employing a Noneigenvalue Formulation", *Journal of Spacecraft and Rockets*, Vol. 20, July–Aug. 1983, pp 363–367.

[Haf-2] Haftka, Raphael T., Martinovic, Zoran N., Hallauer, William L. Jr. and Schamel, G., "Sensitivity of Optimized Control Systems to Minor Structural Modifications", Proceedings of the 26th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics and Materials Conference, Orlando, Florida, April 15–17, 1985, Vol. 2, pp 642–650.

[Haf-3] Haftka, Raphael T., Martinovic, Zoran N. and Hallauer, William L. Jr., "Enhanced Vibration Controllability by Minor Structural Modifications",*AIAA Journal*, Vol. 23, No. 8, August 1985, pp 1260–1266.

[Hal-1] Hale, A.L., Lisowski, R.J. and Dahl, W.E., "Optimizing Both the Structure and the Control of Maneuvering Flexible Structures", Proceedings of the AAS/AIAA Astrodynamics Conference, Lake Placid, New York, August 22–24, 1983.

[Hal-2] Hale, A.L., Lisowski, R.J. and Dahl, W.E., "Optimal Simultaneous Structural and Control Design of Maneuvering Flexible Spacecraft", *J. Guidance and Control*, Vol. 8, No. 1, Jan.–Feb. 1985, pp 86–93.

[Hal-3] Hale, A.L. and Lisowski, R.J., "Characteristic Elastic Systems of Time-Limited Optimal Maneuvers",*J. Guidance and Control*, Sept.–Oct 1985, pp 628–636.

[Hal-4] Hale, A.L., "Integrated Structural/Control Synthesis via Set-Theoretic Methods", AIAA paper 85-0806.

[Han] Hanks, B.R. and Skelton, R.E., "Designing Structures for Reduced Response by Modern Control Theory", Proceedings of the 24th AIAA Structures, Structural Dynamics and Materials Conference, Lake Tahoe, Nevada, May 2–4, 1983.

[Hor] Horta, L.G., Juang, J-N. and Junkins, J.L., "A Sequential Linear Optimization Approach for Controller Design", *J. Guidance, Control and Dynamics*, Vol. 9, No. 6, Nov.–Dec. 1986, pp 699–703.

[IMSL] "IMSL User's Guide", IMSL, Houston, Texas.

[Jua] Juang, Jer-Nan, Ghaemmaghami, Peiman and Lim, Kyong Been, "On the Eigenvalue and Eigenvector Derivatives of a Non-Defective Matrix", 29th AIAA/ASME/ASCE/AHS Structures, Structural Dynamics and Materials Conference, Part 3, Williamsburg, Virginia, April 18–20, 1988.

[Jun] Junkins, J.L. and Rew, D.W., "A Simultaneous Structure/Controller Design Iteration Method", Proceedings of the 4th American Control Conference, Boston, MA., June 19–21, 1985, Vol 3, pp 1642–1647.

[Kho-1] Khot, N.S., Eastep, F.E. and Venkayya, V.B., "Simultaneous Optimal Structural/Control Modifications to Enhance the Vibration Control of a Large Flexible Structure", Proceedings of the AIAA Guidance, Navigation and Control Conference, Snowmass, CO, Aug. 19–21, 1985, pp 459–466.

[Kho-2] Khot, N.S., "Minimum Weight and Optimal Control Design of Space Structures", NATO Advanced Study Institute, Computer Aided Optimal Design: Structural and Mechanical Systems, Troia, Portugal, June 29–July 11, 1986.

[Kir-1] Kirk, D.E., *Optimal Control Theory, An Introduction*, Prentice-Hall, Englewood Cliffs, New Jersey, 1970.

[Kir-2] Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P., "Optimization by Simulated Annealing", Research Report RC 9355, IBM, Yorktown Heights, 1982.

[Kom] Komkov, V., "Simultaneous Control and Optimization for Elastic Systems", Proceedings of the Workshop on Applications of Distributed System Theory to the Control of Large Space Structures, JPL Publication 83-46, ed. by G. Rodriguez, July 1983.

[Lim-1] Lim, K.B. and Junkins, J.L., "Robustness Optimization of Structural and Controller Parameters", *J. Guidance, Control and Dynamics*, Vol. 12, No. 1, Jan.–Feb., 1989, pp 89–96.

[Lim-2] Lim, K.B., "A Unified Approach to Structure and Controller Design Optimization", Ph.D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Va., 1986.

[Lust] Lust, Robert Vernon, "Control Augmented Strucutral Synthesis", Ph.D. Dissertation, UCLA, 1986.

[Man] Manning, R.A. and Schmit, L.A., "Control Augmented Structural Synthesis with Transient Response Constraints", AIAA paper 87-0749.

[Mei] Meirovitch, Leonard, *Computational Methods in Structural Dynamics*, Sijthoff & Noordhoff, Rockville, Maryland, 1980.

[Mes] Messac, A., Turner, J. and Soosaar, K., "An Integrated Control and Minimum Mass Structural Optimization Algorithm for Large Space Structures", Proceedings of the JPL Workshop on Identification and Control of Flexible Space Structures, San Diego, California, June 4–6, 1984.

[Mil-1] Miller, D.F. and Shim, J., "Gradient Based Combined Structural and Control Optimization",*J. Guidance and Control*, Vol. 10, No. 3, May–June 1987, pp 291–298.

[Mil-2] Mills-Curran, William C., "Calculation of Eigenvector Derivatives for Structures with Repeated Eigenvalues",*AIAA Journal*, Vol. 26, No. 7, July 1988, pp 867–871.

[Miu] Miura, H. and Schmit, L.A., "NEWSUMT — A Fortran Program for Inequality Constrained Function Minimization — User's Guide", NASA CR 159070, June 1979.

[Nav] Navid, Mehdi, "Integrated Structural and Control System Design Optimization for Structures Under Dynamic Loading", Ph.D. Dissertation, UCLA, 1986.

[Nel] Nelson, Richard B., "Simplified Calculation of Eigenvector Derivatives", *AIAA Journal*, Vol 14, No. 9, Sept. 1976, pp 1201–1205.

[Nob] Noble, Ben and Daniel, James W., *Applied Linear Algebra*, 2nd Edition, Prentice-Hall, Inc., 1977.

[Ono] Onoda, Junjiro and Haftka, Raphael T., "Simultaneous Structure/ Control Optimization of Large Flexible Spacecraft", AIAA paper 87-0823.

[Pat] Patel, R.V. and Toda, M., "Quantitative Measure of Robustness for Multivariable Systems", Proceedings of the JACC, San Francisco, TP8-A, 1980.

[Pla] Plaut, R.H. and Huseyin, K., "Derivatives of Eigenvalues and Eigenvectors in Non-Self-Adjoint Systems", *AIAA Journal*, Vol. 11, Feb. 1973, pp 250–251.

[Rog] Rogers, L.C., "Derivatives of Eigenvalues and Eigenvectors", *AIAA Journal*, Vol. 8, May 1970, pp 943–944.

[Ros-1] Rosen, J.B., "The Gradient Projection Method for Nonlinear Programming. Part I. Linear Constraints", *J. SIAM*, 1960, pp 181–217.

[Ros-2] Rosen, J.B., "Iterative Solution of Nonlinear Optimal Control Problems",*J. SIAM Control Series A*, 1960, pp223–244.

[Ros-3] Rosen, J.B., "Optimal Control and Convex Programming", *Nonlinear Programming*, J. Abadie ed., New York: John Wiley & Sons Inc., 1967.

[Sal] Salama, M., Hamidi, M. and Demsetz, L., "Optimization of Controlled Structures", Proceedings of the JPL Workshop on Identification and Control of Flexible Space Structures, San Diego, California, June 4–6, 1984.

[Sch] Schmit, L.A., and Farshi, B., "Some Approximation Concepts for Efficient Structural Synthesis",*AIAA Journal*, Vol. 12, No. 5, 1974, pp 692–699.

[Sla] Slater, G.L., "A Disturbance Model for the Optimization of Control/Strucutre Interactions for Flexible Dynamic Systems", AIAA Guidance, Navigation and Control Conference, Minneapolis MN, August 15–17, 1988, pp 57–63; AIAA Paper 88-4058-CP.

[Sta] Starnes, J.H. Jr. and Haftka, R.T., "Preliminary Design of Composite Wings for Buckling, Strength and Displacement Constraints", *Journal of Aircraft*, Vol. 16, No. 8, August 1979, pp 564–570.

[Str] Strunce, R., Lin, J., Hegg, D and Henderson, T., "Actively Controlled Structures Theory", Final Report, Vol 2 of 3, R-1338, Charles Stark Draper Laboratory, Cambridge, Mass. Dec. 1979.

[Van-1] Vanderplaats, G.N., "CONMIN — A Fortran Program for Constrained Function Minimization", NASA TM X-62, 282, August 1973.

[Van-2] Vanderplaats, Garret N., *Numerical Optimization Techniques for Engineering Design: with Applications*, McGraw Hill, 1984.

[Van-3] Vanderplaats, G.N., "ADS — A Fortran Program for Automated Design Synthesis, Version 1.10", 1985.

Appendix A

Results Concerning Matrix Trace Derivatives

For scalar s and matrix A with elements a_{ij}, define the matrix $\left[\frac{\partial s}{\partial A}\right]$ as

$$\left[\frac{\partial s}{\partial A}\right]_{ij} = \frac{\partial s}{\partial a_{ij}} \tag{A.1}$$

Then the following results on the trace operator hold:

$$\frac{\partial}{\partial A}\,\mathrm{tr}[A] = I \tag{A.2}$$

$$\frac{\partial}{\partial A}\,\mathrm{tr}[AB] = \frac{\partial}{\partial A}\,\mathrm{tr}[BA] = B^T \tag{A.3}$$

$$\frac{\partial}{\partial A}\,\mathrm{tr}[A^T B] = \frac{\partial}{\partial A}\,\mathrm{tr}[B^T A] = B \tag{A.4}$$

$$\frac{\partial}{\partial A}\,\mathrm{tr}[B^T AC] = BC^T \tag{A.5}$$

$$\frac{\partial}{\partial A}\,\mathrm{tr}[A^T BA] = (B + B^T)A \tag{A.6}$$

$$\frac{\partial}{\partial A}\,\mathrm{tr}[ABA^T] = A(B + B^T) \tag{A.7}$$

$$\frac{\partial}{\partial A}\,\mathrm{tr}[A^T A] = 2A \tag{A.8}$$

Appendix B

Derivatives of Eigenvectors Associated with Repeated Eigenvalues of Structural Systems

This appendix presents two methods for obtaining eigenvector derivatives of structural systems with distinct eigenvalue derivatives. The first method, due to Mills-Curren [Mil-2], requires only one eigenvalue and its associated complete subspace of eigenvectors, and only the results are given here. The second method is derived here and assumes knowledge of the entire system eigenstructure. For a more general method that does not require distinct eigenvalue derivatives, the interested reader is referred to Juang, et.al. [Jua].

Method 1

The real, symmetric structural eigenproblem (defining K and M as the stiffness and mass matrices respectively) is

$$F_i\phi_i = 0 \qquad \text{for } i = 1,\ldots,n \tag{B.1}$$

where

$$F_i = [K - \lambda_i M] \tag{B.2}$$

with mass normalization

$$\phi_i^T M \phi_j = \delta_{ij} \qquad \text{for } i = 1,\ldots,n \tag{B.3}$$

where δ_{ij} is the Kronecker delta function.

Let the first and second derivatives with respect to some parameter be denoted by $'$ and $''$ respectively. Then differentiating equation (B.1) gives,

$$F_i'\phi_i + F_i\phi_i' = 0 \qquad \text{for } i = 1, \dots, n \tag{B.4}$$

Premultiplying by ϕ_i^T, and using the orthonormality condition (B.3), we can obtain an expression for the eigenvalue derivatives as

$$\lambda_i' = \phi_i^T[K' - \lambda_i M']\phi_i \qquad \text{for } i = 1, \dots, n \tag{B.5}$$

Because the original system is symmetric, an independent eigenvector will exist for every eigenvalue. However, eigenvectors associated with repeated eigenvalues will not be unique, but will form a subspace of the same dimension as the eigenvalue multiplicity, from which any vector will be an eigenvector of that eigenvalue. Equation (B.5) cannot be used to find the derivatives of eigenvalues with multiplicities greater than one until the arbitrary nature of the associated eigenvectors is removed, as the derivatives may not exist for every choice of A.

To remove the arbitrariness in the ψ_j, an eigenvector continuity condition must be imposed to insure that the system eigenvectors remain continuous as the system is perturbed about the condition where the repeated eigenvalues are found. This essentially chooses a unique set of eigenvectors from the subspaces associated with the repeated eigenvalues, any ensures existance of the eigenvector derivatives.

Since the eigenvectors associated with repeated eigenvalues form a subspace, we may write

$$\psi_j = \sum_{\ell \in \Omega_i} \phi_\ell a_{\ell j} \tag{B.6}$$

or

$$\Psi_i = \Phi_i A \tag{B.7}$$

In the above equations, the ϕ_ℓ are the eigenvectors of the repeated eigenvalue λ_i, where $\ell \varepsilon \Omega_i = \{i, i+1, \ldots, i+m_i-1\}$ and m_i is the multiplicity of λ_i. The vectors ψ_j are linear combinations of the ϕ_ℓ, with the coefficients $a_{\ell j}$ to be determined. The columns of Ψ_i contain the vectors ψ_j, the columns of Φ_i contain the vectors ϕ_j, and a_{ij} is the ij^{th} element of A. The matrix Φ_i is the matrix of nominal eigenvectors associated with λ_i, and Ψ_i is the matrix of unique eigenvectors associated with λ_i. Substitution of equation (B.7) into equation (B.1) shows that the ψ_j satisfy the originai eigenproblem for *any* choice of the $a_{\ell j}$.

After imposing the continuity condition, the columns of A and the respective eigenvalue derivatives are found by solving the subeigenproblem

$$[K' - \lambda_i M']\boldsymbol{a}_j = \lambda_i' \boldsymbol{a}_j \tag{B.8}$$

A *unique* matrix A can be found in this manner only if the derivatives of repeated eigenvalues are distinct, i.e. iff

$$\lambda_i' \neq \lambda_j' \quad \text{when} \quad \lambda_i = \lambda_j \tag{B.9}$$

Assuming that this is so, the A matrix then defines a set of unique linearly independent orthogonal eigenvectors for the original eigenproblem (B.1). The derivatives of these eigenvectors can now be found. From this point, assume that the ϕ_i represent the set of unique eigenvectors as found above. Then the eigenvector sensitivity is assumed to have the form

$$\phi_i' = V_i + \Phi_i \boldsymbol{c}_i \qquad \text{for } i = 1, \ldots, n \tag{B.10}$$

where V_i is the vector solution to

$$F_i V_i = -F_i' \phi_i \qquad \text{for } i = 1, \ldots, n \tag{B.11}$$

Since F_i is not of full rank, equation (B.11) cannot be solved by direct inversion.

However. if the appropriate m_i rows and columns are eliminated from F_i along with the corresponding rows from the right-hand side of equation (B.11), then a valid solution for V_i can be found. Guidance for the partitioning of equation (B.11) can be found in the original reference [Mil-2], and will not be reproduced here.

The elements c_{ij} of the vectors $\mathbf{c}_j$ can be found from

$$c_{ii} = -\tfrac{1}{2}\phi_i^T(M'\phi_i + 2MV_i) \qquad \text{for } i = 1, \ldots, n \tag{B.12}$$

$$c_{ji} = \frac{\phi_j^T(K'' - 2\lambda_i' M' - \lambda_i M'')\phi_i + 2\phi_j^T F_i' V_i}{2(\lambda_i' - \lambda_j')} \qquad \text{for } i, j = 1, \ldots, n \text{ and } i \neq j \tag{B.13}$$

Method 2

In this method, we again assume that a unique set of eigenvectors for the eigenproblem have been found as outlined in Method 1 above. However, we now assume the eigenvector derivative is a linear combination of these eigenvectors, in the form

$$\phi_i' = \sum_{j=1}^{n} \phi_j b_{ji} = \boldsymbol{\Phi} \boldsymbol{b}_i \tag{B.14}$$

Substituting equation (B.14) into the mass orthonormality condition (B.3), we obtain

$$b_{ii} = -\tfrac{1}{2}\phi_i^T M' \phi_i \tag{B.15}$$

For any $j \neq i$ such that $\lambda_j \neq \lambda_i$, expressions for the coefficients b_{ij} are well known [Fox], and are

$$b_{ji} = \frac{1}{(\lambda_i - \lambda_j)}\phi_i^T Z_i \phi_i \tag{B.16}$$

where

$$Z_i = [K' - \lambda_i M'] \tag{B.17}$$

To obtain the remaining unknown coefficients, differentiate equation (B.4) again, to get

$$F_i''\phi_i + 2F_i'\phi_i' + F_i\phi_i'' = 0 \qquad \text{for } i = 1, \ldots, n \tag{B.18}$$

Premultiplying equation (B.18) by ϕ_j^T, for j such that $\lambda_i = \lambda_j$, we get

$$\phi_j^T F_i''\phi_i + 2\phi_j^T F_i'\phi_i' = 0 \tag{B.19}$$

since $\phi_j^T F_i = 0$. Substituting equation (B.14) for the eigenvector derivative, we get

$$\phi_j^T F_i''\phi_i + 2\phi_j^T F_i' \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell \right) = 0 \tag{B.20}$$

Consider the first term of equation (B.20):

$$\begin{aligned} \phi_j^T F_i''\phi_i &= \phi_j^T[K'' - 2\lambda_i' M' - \lambda_i M'' - \lambda_i'' M]\phi_i \\ &= [K'' - 2\lambda_i' M' - \lambda_i M''] - \lambda_i''\delta_{ij} \end{aligned} \tag{B.21}$$

Now, consider the second term of equation (B.20):

$$\begin{aligned} 2\phi_j^T F_i' \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell \right) &= 2\phi_j^T[K'' - \lambda_i' M - \lambda_i M'] \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell \right) \\ &= 2\phi_j^T[K'' - \lambda_i' M - \lambda_j M'] \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell \right) \end{aligned}$$

$$
\begin{aligned}
&= 2\phi_j^T[K'' - \lambda_j' M - \lambda_j M'] \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell\right) + \\
&\qquad + 2(\lambda_j' - \lambda_i')\phi_j^T M \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell\right) \\
&= 2\phi_j^T F_j' \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell\right) + 2b_{ji}(\lambda_j' - \lambda_i') \qquad \text{(B.22)}
\end{aligned}
$$

To reduce the first term on the right hand side of equation (B.22), substitute equation (B.14) into the transpose of equation (B.4), to get

$$
\phi_j^T F_j' = -\left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell\right)^T F_j \qquad \text{(B.23)}
$$

Since $\lambda_j = \lambda_i$, then $F_j = F_i$, and

$$
\phi_k^T F_i = \begin{cases} 0 & \text{if } \lambda_k = \lambda_i \\ (\lambda_k - \lambda_i)\phi_k^T M & \text{if } \lambda_k \neq \lambda_i \end{cases} \qquad \text{(B.24)}
$$

Therefore,

$$
\begin{aligned}
2\phi_j^T F_j' \left(\sum_{m=1}^{n} b_{mi}\phi_\ell\right) &= -2\left(\sum_{\ell=1}^{n} b_{\ell j}\phi_\ell\right)^T F_i \left(\sum_{m=1}^{n} b_{mi}\phi_m\right) \\
&= -2\sum_{\substack{\ell=1 \\ \ell\notin\Omega_i}}^{n} b_{\ell j}(\lambda_\ell - \lambda_i)\phi_\ell^T M \left(\sum_{m=1}^{n} b_{mi}\phi_m\right) \\
&= -2\sum_{\substack{\ell=1 \\ \ell\notin\Omega_i}}^{n} b_{\ell j} b_{\ell i}(\lambda_\ell - \lambda_i) \qquad \text{(B.25)}
\end{aligned}
$$

where $\Omega_i = \{\ k :\ \ \lambda_k = \lambda_i\ \}$. Since we already have expressions for $b_{\ell j}$ and $b_{\ell i}$ for $\ell \notin \Omega_i$ given in equation (B.16), then equation (B.25) becomes

$$
2\phi_j^T F_i' \left(\sum_{\ell=1}^{n} b_{\ell i}\phi_\ell\right) = -2\sum_{\substack{\ell=1 \\ \ell\notin\Omega_i}}^{n} (\lambda_\ell - \lambda_i)\left[\frac{(\phi_\ell^T Z_j \phi_j)\ (\phi_\ell^T Z_i \phi_i)}{(\lambda_j - \lambda_\ell)\ (\lambda_i - \lambda_\ell)}\right]
$$

$$= \ 2 \sum_{\substack{\ell=1 \\ \ell \notin \Omega_i}}^{n} \frac{(\phi_\ell^T Z_j \phi_j)\,(\phi_\ell^T Z_i \phi_i)}{(\lambda_i - \lambda_\ell)} \tag{B.26}$$

Substituting equation (B.26) into equation (B.22), then equations (B.21) and (B.22) into equation (B.20), we obtain

$$\phi_j^T[K'' - 2\lambda_i' M' - \lambda_i M'']\phi_i - \lambda_i''\delta_{ij} + 2b_{ji}(\lambda_j' - \lambda_i') +$$

$$+ 2 \sum_{\substack{\ell=1 \\ \ell \notin \Omega_i}}^{n} \frac{(\phi_\ell^T Z_j \phi_j)\,(\phi_\ell^T Z_i \phi_i)}{(\lambda_i - \lambda_\ell)} \ = \ 0 \tag{B.27}$$

Forcing $i \neq j$, and rearranging, gives the required coefficient b_{ij} for $i, j \notin \Omega_i$ as

$$b_{ji} = \frac{1}{(\lambda_j' - \lambda_i')} \left\{ \sum_{\substack{\ell=1 \\ \ell \notin \Omega_i}}^{n} \frac{(\phi_\ell^T Z_j \phi_j)\,(\phi_\ell^T Z_i \phi_i)}{(\lambda_i - \lambda_\ell)} + \tfrac{1}{2}\phi_j^T[K'' - 2\lambda_i' M' - \lambda_i M'']\phi_i \right\} \tag{B.28}$$

Note that differentiating the M-orthogonality condition (B.3) between ϕ_i and ϕ_j gives

$$\phi_j^{T\prime} M \phi_i + \phi_j^T M \phi_i' = -\phi_j^T M' \phi_i \tag{B.29}$$

Substitution of equation (B.14) into equation (B.29) gives

$$b_{ij} + b_{ji} = -\phi_j^T M' \phi_i \qquad \text{for } i = 1, \ldots, n \tag{B.30}$$

Equation (B.28) satisfies equation (B.30), as would be expected. Also note that forcing $i = j$ in equation (B.27) gives the second derivative of the eigenvalues as

$$\lambda_i'' = 2 \sum_{\substack{\ell=1 \\ \ell \notin \Omega_i}}^{n} \frac{(\phi_\ell^T Z_i \phi_i)^2}{(\lambda_i - \lambda_\ell)} + \phi_i^T [K'' - 2\lambda_i' M' - \lambda_i M''] \phi_i \tag{B.31}$$

*U.S.Government Printing Office: 1990—748-056/24469